For Reference

NOT TO BE TAKEN FROM THIS ROOM

Ex libris
universitatis
albertaensis

QUAECUMQUE VERA

BRUCE PEEL SPECIAL COLLECTIONS LIBRARY

UNIVERSITY OF ALBERTA LIBRARY

REQUEST FOR DUPLICATION

I wish a photocopy of the thesis by

Kearsley Greg. P (author)

entitled A study of learner control in computer-based Instruction

The copy is for the sole purpose of private scholarly or scientific study and research. I will not reproduce, sell or distribute the copy I request, and I will not copy any substantial part of it in my own work without permission of the copyright owner. I understand that the Library performs the service of copying at my request, and I assume all copyright responsibility for the item requested.

Date	Name and address	Pages copied	Signature
August 19/91	James Murtagh 27 MacEwan Park Rd N.W. CALGARY	PP. 1, 2	
		PP. 4, 5, 6, 7	
		P. 10	
		P 15	
		P. 26	
		PP. 31, 32, 33	
		P. 71	
		PP 75, 76	
		Title Page.	

Digitized by the Internet Archive
in 2022 with funding from
University of Alberta Library

https://archive.org/details/Kearsley1978

THE UNIVERSITY OF ALBERTA

A Study of Learner Control In Computer Based Instruction

By

© Greg P. Kearsley

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF DOCTOR OF PHILOSOPHY

IN

EDUCATIONAL PSYCHOLOGY

EDMONTON, ALBERTA

FALL 1978

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled A Study of Learner Control in Computer Based Instruction submitted by [illegible] in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Educational Psychology.

....
Supervisor

....

....

....

....
External Examiner

Date....June [illegible]....

ABSTRACT

This thesis represented a study of the issue of Learner Control (LC) in Computer Based Instruction (CBI). The LC issue is a complex one and involves a number of different aspects. This includes the problem of who is to have the power of control over the content and direction of instruction. Another aspect of the issue is the question of whether instruction can be better adapted to individual differences through the instructor's selection of parameters or through the student's selection. A third component of the issue has to do with the instructional design considerations of task sequencing, instructional strategies, media selection, etc. These three major aspects as well as the results of past research on LC were brought together in the formulation of a theoretical framework for LC.

This framework was used as a basis for a study of LC in the context of a computer-based course to teach the programming language APL to high school and university students. The study provided some students with full control over concept or topic sequencing, instructional mode, amount of practice, difficulty level, and a number of other features, other students with optional control, and a third group with no control.

The total completion times, average number correct, learning rate, and attitude measures were compared among the three LC conditions. Measures of internal/external locus of control and programming and mathematical background were also compared among the groups. The overall results of the study suggested that increased degrees of LC as implemented had no beneficial effects on the performance or attitudinal measures in the context of tutorial APL instruction. Nor was any relationship found between the personality or preskill measures and the use of LC.

The conclusions of the study identified inadequacies in the theoretical framework and discussed the role of LC in the present and future development of CBI. It was suggested that the tutorial mode of instruction tends to mask the potential benefits of LC and that these benefits may be more obvious in a problem solving or socratic type of CBI where the learner must play a more active role during instruction. It was also suggested that LC can not be properly implemented until CBI systems have the capability to qualitatively represent the conceptual structure of the student in terms of the skills or components of the task or subject matter.

ACKNOWLEDGEMENTS

In addition to my committee members, I would like to thank J. Anderson, W. Ebersberger, K. Fenske, J. Huntington, T. McGuire, C. Montgomerie, N. McGinnis, and the 1500 operators for their logistic, intellectual, moral, or spiritual help along the way.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. The Issue of Learner Control (LC).

Computer Assisted Instruction (CAI) and Computer Assisted Learning (CAL) are two terms commonly used to describe the instructional use of the computer. The difference between these two terms is not trivial. Rather, it reflects quite different (and often antagonistic) approaches or orientations to the use of computers in education. Those who advocate the CAL position feel that the computer should be a tool of the student -- a tool which broadens the scope or depth of learning activities. Thus, the student should learn how to use computers in order to utilize large data bases, perform calculations, analyze data, construct simulation models or even create games. Luehrmann (1972) states this viewpoint as follows:

> Computing constitutes a new and fundamental intellectual resource. To use that resource as a mere delivery system for instruction, but not to give a student instruction in how he might use the resource himself, has been the chief failure of the CAI effort. What a loss of opportunity if the skill of computing were to be harnessed for the purpose of turning out masses of students who are unable to use computing. (p. 44)

Other vocal supporters of CAL have been Papert and his crusade for the use of the computer language LOGO (e.g., Papert & Solomon, 1972), Dwyer (1974) and his SOLOworks project, and Pelee (1974) who has argued for the use of

"glass boxes" in teaching APL.

In contrast, those who hold the CAI perspective conceptualize the computer as a tool of the instructor -- a tool which can increase the effectiveness of teaching by virtue of controlled and monitored delivery. In this conceptualization, the basic theory associated with CAI is one of control or adaptation; how to optimize the student's trajectory through a curriculum to produce the best performance in the least time. The best examples of this viewpoint are the drill & practice programs developed at Stanford in arithmetic and reading (Atkinson, 1972; Suppes & Morningstar, 1972). As Atkinson (1976) says:

> Some have argued that any attempt to devise optimal strategies is doomed to failure, and that the learner is the best judge of appropriate instructional actions. I am not sympathetic to a learner-controlled approach to instruction because I believe its advocates are trying to avoid the difficult but challenging task of developing a theory of instruction. (p. 106)

Pask (1975) and Landa (1974) are other researchers who have approached CAI from a control theory or "cybernetic" perspective.

The distinction between CAL and CAI reflects a basic conflict in beliefs or educational philosophies. In fact the CAL - CAI conflict is the most recent version of an enduring

power struggle in education over who should control the course and nature of learning. Classical thinkers such as Plato, Socrates, Rousseau, and Locke all took positions on this issue. Contemporary statements of the two sides are given by Rogers (1972) in Freedom to Learn and by Skinner (1968) in The Technology of Teaching. Furthermore, learner control is a major aspect of many other educational controversies or innovations such as Summerhill and the "free school" movement,deschooling society (Illich, 1970) or dialogics (Freire, 1972). Thus the issue of learner control is not unique to computer-based education but is a general issue in educational theory.

Advocates of CAI see the use of the computer by the student to solve problems or search data bases as particular types of instructional modes, e.g., simulations, inquiry, drill & practice, tutorial, etc., which are matched to the requirements of the task or subject matter. CAL advocates, on the other hand, consider CAI as one, somewhat limited, use of the computer as a learning tool.[1] When CAL and CAI are seen as dichotomous positions, there is little hope of making any useful progress in resolving differences. The difference between CAI and CAL is the degree and type of learner control permitted. Thus the amount of student control can range over a number of different parameters

1. Indeed, Papert & Solomon (1972) suggest CAI as only one of 20 possible topics in their fifth grade computer science course.

involved in learning. These parameters include pacing, sequencing, amount of practice, difficulty, media selection, learning styles, and experience with learner control. The useful question to ask is what parameters are better left in the hands of the student and which in the hands of the instructor in terms of student performance or satisfaction. In other words, the question is not really who is to control instruction, but who controls <u>what</u> and <u>how</u>. The present thesis addresses this question.

Part of the learner control issue derives from the pejorative connotations of the word "control". It is often interpreted in the sense of coercion or manipulation. However, control actually has a generically neutral meaning, i.e, the influence of one system on another. In the case of instruction we usually think of the influence of a "teacher" on a "student". However, a student influences his/her own learning, i.e. self-control. As Landa (1977, p.8) points out: "the end goal of instructional control is to produce in the student the ability for self-control, which includes as its component, the ability for independent goal setting". As this quote suggests, teacher and student control are progressive stages of the instructional / learning process. As the student continues with formal education, the instruction becomes more self-directed and less teacher directed. However, the rate of this progression will depend upon the age, experience, and motivation of the learner. A further remark by Landa outlines the problem of power

involved in this transition: "Each increase in the development of self-control abilities must be followed by a corresponding weakening of the external control and a handing over of the corresponding functions of control to the student himself" (p. 8). In other words, the teacher must be willing to accept the proposition that students can learn better on their own than under the direction of the teacher.

Much of this same problem also has to do with the evaluation or standards of education. It can be argued that there are certain subject matters or instructional contexts in which allowing student control is undesirable in terms of maintaining necessary educational standards. Thus in the case of children learning to read, medical students or pilot training, for example, it is argued that mastery (or some minimal level of performance) must be guaranteed and that permitting the student to control the selection or sequencing of topics or the amount of practice would jeopardize this. This argument is basically about who has the power to set the standards or criteria of instruction, i.e., whether the responsibility will be in the hands of the student or instructor.

A second facet of the learner control issue, confounded with the first, has to do with the individualization of instruction, i.e., the mechanism for provding each student with a unique learning experience tailored to their

particular learning styles, preskills, or personality type. One approach to individualization (the adaptive approach) is based upon the idea of using the student's past response history in a task to alter the course or nature of future instruction. This approach may involve changing difficulty level, mode of instruction, or amount of practice based upon recent or cumulative performance. The Stanford drill & practice programs are examples of this type of individualization. Another approach to individualization in CAI involves using aptitude and personality measures (the trait approach). To use this approach one must be able to associate individual difference measures with particular instructional parameters. An enormous body of literature on Aptitude Treatment Interaction (ATI) research exists (reviewed in Cronbach & Snow, 1977) and there have been specific proposals about how to relate individual differences measures to instructional parameters (e.g., Stolurow, 1972). However, Merrill (1975) has argued that:

> A given person never has exactly the same configuration of these momentary aptitudes twice in his life. Consequently, the search for the interaction of stable trait aptitudes and fixed treatments is never likely to be of instructional value. (p. 221)

Instead Merrill argues that individuals should be given a means of adapting the learning materials to their aptitudes and styles and that such learning strategies or tactics should be changeable during the course of learning. Merrill

is arguing against both the adaptive and the trait theory approach since the mechanism for is put individualization in the hands of the instructor rather than the student. Is Merrill right? Is this really the best way to individualize instruction across the entire range of individual differences related to learning? Or is the adaptive or trait approach the most appropriate way to individualize instruction? This thesis will also consider this aspect of the LC issue.

There is yet one more side to the LC issue which has to do with the task analysis component of CAI. A great deal of literature exists in areas such as curriculum and instructional design which is concerned with the optimal arrangement or delivery of instruction. A good example is the sequencing of material to be learned. On the basis of much past research, there are many recommendations made regarding the optimal way to sequence a particular subject matter or task (e.g., Briggs, 1968; Posner & Strike, 1976; Tennyson, 1972). According to the instructional design perspective, there should be optimal ways to organize and present a particular subject matter or task and hence allowing the student to alter these would always result in suboptimal instruction. Clearly, there is no acknowledgement of individual differences here. So the question which arises is whether , in fact, there can be an optimal instructional design for all students and what is lost if the student is allowed to modify this. This thesis will explore this aspect

of learner control.

A discussion of learner control within the intructional design perspective must be related to the type and level of learning involved, the nature of the subject matter or task, the capabilities of the student, and the instructional strategies involved. With respect to types and levels of learning, problem solving will tend to require learner control of one kind whereas learning facts or terminology (i.e., knowledge or comprehension) will tend to necessitate another. Problem solving will require control over the instructional procedures or modes while the learning of facts will involve the sequencing of concepts. The same contrast is true of instructional strategies. A simulation strategy, almost by definition, involves complete learner control while a drill & practice strategy provides little room for student choice. In the simulation strategy, the learner essentially determines how instructionally effective the program is by the nature of the input he/she provides. On the other hand, in drill and practice, control over the level of difficulty is probably the most important. Similarly, different types and levels of subject matter or tasks will allow for more or less learner control along certain dimensions by virtue of their inherent structure. Thus, arithmetic and topology are both mathematics but would involve different structures and control options.

As the preceding discussion suggests, the learner

control issue is a complex one with many different interconnected strands. The three main strands outlined above are (1) the instruction and/or learning processes aspect; (2) the individualization aspect; and (3) the instructional design aspect. Each of the aspects leads to a slightly different formulation of the learner control issue which implies that any theoretical explanation of learner control will have to integrate the essential ideas from three different research traditions (i.e., learning, individual differences, and instructional theory). The theoretical significance of this study lies in this integration. The practical importance pertains to the design of instructional software and hence the type of courseware which is and can be developed. The ultimate success of computer-based instruction as an educational methodology may depend upon a greater understanding of learner control. It is hoped that the present study makes some contribution to this practical problem.

II. Previous research on LC in CAI.

Learner Control has become a well studied topic in instructional psychology over the last decade. Reviews of the literature (e.g., Geis, 1976; George, 1976) have summarized the results of these past studies. The following literature review focuses on LC research involving CAI.

The first study to be discussed does not involve CAI but is relevant to later studies. Issing & Eckart (1973) compared teacher versus learner control in the context of a course on computer fundamentals. The teacher-controlled group heard an audio tape prepared by one of a number of teachers along with prepared illustrations whereas the student-controlled group were given the illustrations and allowed to interrogate a teacher via a teletype. The students were 54 undergraduates. A pretest was used to eliminate anyone who already knew something about computers. The results of the study were:

(1) The teacher-controlled sequences as composed by different teachers were very similar.

(2) The learner controlled sequences (i.e., types and patterns of questions asked) were very similar.

(3) The teacher-controlled and learner controlled sequences were different.

(4) Students prefered the learner controlled method.

(5) The teacher-controlled method was more effective for factual knowledge.

(6) The learner controlled method was more effective when transfer of learning was involved.

Many of the results were based upon hypotheses generated from previous research (e.g., Mager & Clark, 1963; Campbell, 1964). This study is notable because most of these findings serve as starting points for subsequent research.

Judd, O'Neil, and Spelt (1974) provide a review of LC research which covers both PI (Programmed Instruction) and CAI instruction. Table 1 gives a summary of nine studies done with LC in CAI which are discussed by Judd et al. (most of these studies are unpublished technical reports; the full citations are given in Judd et al.). As the Table indicates, the outcomes of LC research have been mixed with no consistent positive or negative findings. (However, only one of the nine studies reports a negative outcome with LC; the rest are neutral or positive.) Judd et al. suggest that this lack of consistency may stem from a number of methodological problems; failure to account for individual differences in reaction to LC; lack of adequate instruction on the use of the LC features; lack of appropriate dependent measures (e.g. affective); and finally the lack of consensus on what constitutes LC itself. They also point out that studies which restrict the range of material available essentially eliminates the need for learner control.

The study conducted by Judd et al. was an attempt to remedy some of these deficiencies. They attempted to gauge the value of LC by measuring its effect on the use of a facilitative variable. The subject matter was a 2 hour course on the identification of edible plants and the subjects were 162 undergraduates. The LC option provided access to a presumably facilitative treatment (mnemonic aids). The design included two control groups: one which received no LC option at all, and the other which received the facilitative treatment (mnemonics). They also varied the degree of instruction received on the LC (brief and extended instructions). Their dependent measures were number of errors, number of LC requests, and responses to a state anxiety measure. Individual difference measures of task specific memory, locus of control, and achievement-independence were also obtained.

Unfortunately, the design was unsuccessful because the mnemonics had a debilitating rather than facilitating effect for most subjects. Thus providing access to mnemonics via LC did not have a beneficial effect on either performance or anxiety measures. Judd et al. attributed this unexpected result to the heavy memory load necessitated by the mnemonics and also the facilitative effect of visual aids (not under LC control). The effect of the extended rather than brief instructions on LC were to increase the number of LC requests in the first instructional segment. The relationships between the individual difference variables

were unclear due to the failure of the LC manipulation.

However, a study by White & Smith (1974) found a relationship between personality dimensions and learner control. The study involved 192 elementary/secondary science education students learning about the use and writing of behavioral objectives. The Myers-Briggs Personality Type Inventory was used to identify introverted, extroverted, sensing, and intuitive personality types. Four learner control groups differed in their control over:

i. whether or not the objectives were shown,

ii. comments on preferences and errors were made,

iii. recommendations for next module were made,

iv. the sequence of modules.

One group had control over i,ii,iii,iv; the second group had control over i, ii, iii; the third group had control over i,ii; and the fourth group had no control over these parameters. The dependent measures were total correct/incorrect, response time, total time, and attitudes towards the subject matter and CAI. The analysis of the data was made in terms of the four personality types. The results indicated a number of interactions between personality types and LC groups. As more control was given to the intuitive types, they became less satisfied with the subject matter and CAI whereas the sensing types became more satisfied. Both the extroverts and introverts tended to perform more error-free when they followed a fixed sequence, however the tendency was stronger for extroverts. This study provides

evidence for the importance of personality characteristics in the effects of learner control.

Atkinson (1972) studied the effects of LC with university students in a drill & practice program for learning German-English vocabulary. Atkinson compared the performance of the students using four different strategies for the sequencing of word selection: random order, student selection, optimal strategy based upon an assumption of equal item difficulty, and optimal strategy based upon an assumption of unequal item difficulty. The optimal strategy was based upon the predictions of a mathematical learning model for maximizing the number of correctly remembered words. The results of these four strategies showed that the optimal selection rule based upon unequal item difficulty was best in a 2 week delayed retention test followed by the student selection strategy which was better than either the random or optimal based upon equal difficulty strategies. However, in terms of the number of correctly remembered words during the learning trials, the random order and optimal order based upon equal difficulty strategies were better than the student selection or optimal based upon unequal difficulty strategies. Atkinson suggests that in the student selection strategy, students tended to practice items they didn't know hence reducing the number they got correct while learning but improving their delayed recall of the words. This conjecture may explain why the LC group showed poorer or slower learning on the basis of immediate

testing but improved learning in tasks which involve longer retention.

Fisher et al. (1975) investigated the effects of LC with 4th and 5th grade pupils on the Stanford arithmetic drill & test program. A yoked LC-control group design was employed so that each pupil in the control group received essentially the same instruction as the LC pupils. The LC options allowed selection of problems and control over the level of difficulty. The dependent measures were the degree of engagement (measured by behavioral observations of the ease of distraction from the task by toys) and a locus of control measure in which control was defined as the attribution of outcomes to factors that can be changed (as opposed to fate or luck). Their findings were that LC students showed significantly greater task engagement and more self-directed attribution on the locus of control measures. An interesting outcome was that the yoked control group completed substantially more problems than the LC group. Within the LC group there were two distinct and consistent choice patterns; children who consistently chose the easier problems (thus maximizing their performance scores) and children who consistently chose harder problems (hence minimizing their performance scores). This result reveals that some students can deliberately lower their performance while using LC.

This last finding is supported by research done by Pask

& Scott (1972) and extended by Pask (1976). In these studies, the subject matter was an artifical zoological taxonomy ("Clobbits" and "Gandlemullers"). Individuals are provided with different types of information available and must learn by selecting certain types of information. Pask classified learners into two broad learning styles, serialists and holists. Serialists learn by means of string-like concept links; they assimilate lengthy sequences and are intolerant of irrelevant information. Holists learn material as a whole; they develop an interrelated system of principles which contain irrelevant information. Pask describes these two learning styles in considerable detail as well as their operational referents. Pask & Scott demonstrated that students who were matched on the type of information available during their learning task and their serialist/holist learning style learned more effectively than those who were mismatched.

Lahey & Crawford (1976) investigated the effects of LC strategies in the U.S. Navy basic electronic/electricity curriculum. Their study involved TICCIT (Timeshared Interactive Computer Controlled Instructional Television) and allowed students to select the mode of learning (rule, example, or practice) as well as changing the level of difficulty. They reported the frequency of use of each of the 16 possible combinations of rule, example, and practice. The results indicated that students had a strong preference for a rule-example-practice strategy followed by an example-

practice strategy. They also found no difference in the order of the strategies used as a function of the length of time spent on the course or changes in conceptual difficulty and hence concluded that familarization with content and difficulty level had no influence on the strategies selected. In other words, the results of this study suggest that when students are provided with a number of possible strategies to use, they tend to rely on one particular strategy.

A subsequent study by Lahey (1978) using the same curriculum addressed the question of whether students would do better if they were given advice as to what lesson content or lesson mode to do next. The performance of three groups was compared: a group with complete learner control, a group in which an arrow suggested the next content/mode to select, and a group which received the content/mode without any choice. There were no significant differences in performance between the three groups. On the basis of the results of this study as well as many previous studies, Lahey concluded that LC had no significant pedagogical advantages over adaptive programmed control.

A study by Fredericks (1976) also used the electronics curriculum with Naval trainees. In this study, performance of a group of trainees whose amount of practice was determined by an adaptive branching algorithm was compared to a group who had control over the amount of practice. The

results indicate that the LC produced large time savings, less variability and higher scores. The data supported the conclusion that a knowledgeable student can realistically estimate when he/she has mastered the practice material.

O'Neal (1977) examined the conclusion of the Fredericks study. He gave college students varying degrees of control over amount of practice and when to take tests in the learning of a hierarchically organized set of APL concepts. Students were given the option to take tests when they wanted or taking the tests after certain practice conditions were met. It was hypothesized that students who had complete control over practice would not achieve the same degree of mastery as those who had to meet pre-specified mastery conditions. The results showed that there was no difference between the groups. It was also hypothesized that learners would become more efficient at using LC over time and that the LC would result in more positive attitudes. These hypotheses were not supported either. O'Neal concluded that students used learner control in such a way as to eliminate differences between the different practice groups. This suggests that LC can be used to compensate for poor instructional design (and hence would not be a factor upon which groups could be discriminated when the instructional design is optimal).

The final study to be reported is a comprehensive one by Seidel et al. (1978) which investigated learner control

of sequencing in a 30 hour course teaching the programming language COBOL. The study investigated four sequencing variables, three concerned with control over review and remedial activity and the other with the sequencing of topics. They measured student entry characteristics (including ability scores and programming aptitude), learning strategies in terms of LC usage and circumstances, and performance measures (times, quiz scores, programming errors). They also measured the student's level of aspiration (LOA) before the quiz for each topic. They analyzed the results in terms of high and low performance (using a discriminant analysis) and found that high and low performers differed in their use of learner control as well as the LOA measure; however, these differences were not uniform for any particular individual across all instructional modules. They concluded that expectations play an important role in performance whether indicated by self-assessment or by the learning system.

These past studies of LC in CAI suggest the following conclusions:

1) LC may be better or worse in terms of performance measures than optimized or adaptive programming depending upon the appropriateness of the parameters under control.

2) LC options tend to produce more favorable affective reactions from students, both towards the subject matter and CAI.

3) When LC options are available, students may use them in a

restricted way and this is related to the degree of instruction and practice received on LC features.

4) LC may cause students to become more deeply engaged or involved in the subject matter and hence lead to increased learning (as measured by delayed retention tests)

5) Individual differences in motivation, achievement, personality types, and learning styles will influence the use and effects of LC.

Since the effects of LC have been found to be equivocal and varied across previous studies, it seems important that subsequent research be evaluative in nature, i.e., investigate LC as it specifically applies to particular subject matters and student groups. Furthermore, much of the past research has been conducted in the absence of any explicit theoretical framework. It would seem that future studies which are developed in the context of a conceptual framework would contribute more to the understanding of LC. The next chapter develops a rule-based theoretical basis for LC which is used in the design of an evaluation study discussed in the remainder of this thesis.

III. A Theoretical Basis for Learner Control.

In concluding their review of past research on LC, Judd et al. (1974) point out that there exists no real consensus on what learner control is. Learner control may refer to control over a wide variety of variables including content area, sequencing of instructional units, pacing, review or help, level of detail or difficulty, or mode of instruction. The lack of systematic definition is not too surprising because almost all learner control research is carried out on an ad-hoc basis and not on the basis of any theoretical framework. The present chapter attempts to outline a theoretical framework for learner control which may provide a more systematic context in which to understand LC. This framework incorporates the ideas of many educational psychologists such as Ausubel, Bruner, Gagne, Scandura, and others. Furthermore, it tries to integrate the traditions of learning theory, individual differences, and instructional design as they pertain to LC.

The basis for the theoretical framework begins with the idea that the root of the learner control issue rests with educational goals or purposes -- those belonging to the student and the teacher. Those of the teacher presumably reflect the requirements and demands of parents, business, and society in general. These include goals such as general literacy, an appreciation of cultural heritage, certain inter-personal skills, etc. The goals of the student reflect

personal achievement, prestige, social recognition, and so on. While to a great extent the goals of teacher and student will overlap, there are some goals which will not be shared by both. Differences in educational goals are the basic level from which the learner control issue originates.

Goals are ultimately translated into instructional objectives (albeit often very vaguely formulated) which define what is to be taught and how. As usually defined, an instructional objective has three main components: (a) the behavior or action to be performed by the student, (b) the conditions under which the behavior is to be elicited and evaluated, and (c) the criteria or standard of behavior to be achieved to demonstrate mastery of the task or skill. Instruction designed to satisfy these three components must include certain macro or microstrategies [2] to produce the desired behaviors, the particular content and context which specifies the conditions under which the behavior is elicited, and sufficient practice to achieve the desired standard. Differences in goals between teachers and students (and also within each group) will be transmitted in terms of these three components. Thus, these three aspects of instruction-- content, strategies and standards-- will be the major dimensions over which teacher/learner control can

---

2. A macrostrategy is a global type of instructional method, i.e., drill & practice, tutorial, group discussion, inquiry, simulation, etc., while a microstrategy is the fine grained techniques such as shaping, cueing, prompting, chaining, etc.

vary.

Content variables include those associated with decisions about what material should be included, excluded or emphasized and also the inherent sequencing and organization of this material. Instructional strategies cover the modes (e.g., inductive/deductive), rates, step and task sizes or use of various delivery media. The standards include the amount of learning required to master a subject or task, the type of evaluation, and the frequency of reviews. The type of evaluation would include simulation, multiple choice, etc., as well as considerations of levels (e.g., according to Bloom's taxonomy). Many variables (e.g., difficulty) will affect content, strategies, and standards. Because of this, control over one aspect of instruction such as content will also affect strategies and standards.

We can conceive of content, strategy and standards as rule-based phenomena.[3] The rules inherent in a subject matter or task are those that define basic concepts (e.g., "things with these attributes are called x"), procedures (e.g., "to get x, you do this"), and the operations which define relationships between concepts and procedures (e.g., "y is the result of doing this to x"). The rules involved in strategies may be algorithms or heuristics designed to

3. Rules have become a major explanatory construct in cognitive psychology. For explications, see Scandura (1972) or Segal & Stacey (1976).

facilitate the learning of the rules of a subject matter or task. For example, the microstrategy of prompting is a rule which suggests that when there is a low probability of a learner making a spontaneous correct response, then some hint or stimulus should be provided which will ensure that the learner makes a correct response. While many of the rules which comprise instructional strategies are content or task independent (such as prompting), many are strategies for a particular content. For example, there are many strategies used in the teaching of arithmetic with respect to how to do carrying, the format of the problems, etc.

The rules which relate to achieving a mastery criterion are those concerned with assessing whether something has been understood, i.e., rules which help "debug" knowledge. A typical rule for assessing whether something has been understood is to try to paraphrase it (in the case of fact or concept) or generate a novel result (in the case of a procedure). The learning of every subject matter or task produces characteristic errors and a knowledge of these is an important aspect of composing successful instructional strategies for that subject matter.

Learning a subject matter thus consists of learning a set of rules and their interrelationships.[4] Because of the

---

4. In fact, it can be argued that the integration of interrelationships between concepts is the more difficult and most important aspect of learning.

inherent logic of a subject matter, there will almost always be some hierarchical structure which demands that some concepts or rules be learned before others (Gagne, 1968). Thus, a student must learn to count before learning addition, or learn the basic vocabulary of a language before composing sentences in it.[5] However, because individuals differ in their prior background of rules and concepts, each individual is likely to construct a somewhat different set of relationships for a given subject matter. This is depicted abstractly in Figure 1. The Figure shows a graph in which nodes represent concepts or rules and the arcs represent inter-relationships between them. The subgraph shown with dashed lines represents the conceptual structure of the subject matter for student "A" while the subgraph with the dotted lines indicates the understanding of the subject matter for student "B". While they have mastered almost the same set of concepts or rules, they have formed many different inter-relationships. Thus, both students have learned a common subject matter but they have constructed their own individual conceptual structures.[6] For example two

---

5. Actually, the word "must" is too strong here. It would be possible, though very arduous, to learn to add without learning to count (say by memorizing combinations of symbols) or to learn complete sentences one by one and not understand their individual constituent words.

6. Figure 1 depicts a rather extreme case. If two students have similar pre-entry skills and educational background, it is likely that there would be considerable overlap in the relationships formed among concepts. In the case of two students with very different pre-entry skills and perhaps socio-cultural backgrounds, differences in conceptual structures (in the sense of different relationships between the same concepts) are much more likely.

students could learn the concept of statistical significance in terms of a strictly numerical comparison rule (i.e., if higher than critical value, reject), while another student could learn it in terms of critical regions on a frequency distribution or a confidence interval. Both of these students would have a grasp of the concept but they differ in how the concept is inter-related to other statistical concepts they know.

This relatively simple point makes a profound difference to a theory of instruction, for it suggests that learning depends upon the reconstruction of the subject matter by the learner. The constructive nature of learning is a major theme of cognitive theory as espoused by Bruner, Piaget, Neisser, and most information processing theorists. The important implication of the constructive view is that learning must be a cognitively active process on the part of the learner and that student involvement is essential to building a conceptual structure. It follows that the student should be given a means of organizing new knowledge in an appropriate way. The constructive view provides a theoretical rationale for an instructional system which allows the student to directly control the acquisition of concepts or rules.

Reconstruction will be facilitated if the concepts/ rules which comprise a subject matter are presented in a manner such that they can be easily reorganized by the

learner. The nature of the instructional process will obviously effect the way that this reorganization takes place. Thus Figure 1 could also represent two different conceptual structures involving the same concepts for the same learner which could have resulted from different instructional methods.[7] There are a number of theoretical ideas which are relevant here. The first is Ausubel's (1968) notion of "advanced organizers". Ausubel argues that the learning of meaningful information can only occur when there exists some sort of conceptual scaffolding upon which subsequent learning can be built. By providing the student with a concise but rich organizer at the beginning of a lesson, the necessary concepts or rules are available to assimilate the details of new concepts or rules. The second relevant idea is Bruner's (1966) proposal for 'spiral curricula' in which successive instruction elaborates or refines previous material. Thus each turn of the 'spiral' provides further depth or scope to a previous concept or rule. Similar to Bruner's spiral curriculum idea is the "web teaching" proposal of Norman (1976). In web teaching, material must be presented in a fashion which allows the student to develop some semantic framework for relating the pieces of information to each other and then continues to present material which can be related to this framework. For example, in a statistics curriculum, one might teach the

---

7. This point is discussed and documented by Mayer (1977) for the concepts of sequencing, order and organization.

general differences between parametric and non-parametric tests and then present individual pairs of tests which illustrate in detail these differences. Essentially, this suggests a "top-down" approach to presenting material. The implication of the web teaching idea is that the curriculum organization should be designed so that such manipulation by the student is facilitated.

Another relevant idea is Gagne's notion of learning hierarchies. Gagne's task analyses of many subject areas (see, for example, Gagne & Briggs, 1974) reveals a hierarchical arrangement of subskills which appear to represent the optimal organization of instruction for a particular task. It is possible to think of these subskills as representing various levels or types of rules.[8] The first-order rules are those inherent in the subject matter itself, e.g., arithmetic rules, spelling rules or grammatical rules. At the next level are rules about learning the subject matter as a whole. At a third level are rules concerning learning in general. Thus, if one were learning geography, a first-order rule might be: "temperature generally declines as altitude increases"; a second-order rule might be: "draw maps to understand the spatial relations"; and a third-order rule might be:

8.According to Gagne, there would be rules for distinguishing certain classes of attributes (discrimination and concept learning), rules for finding solutions (problem solving) and rules for deducing/inducing facts (principle learning).

"regular review leads to less cramming" (this rule is not specific to geography). The third level would correspond to "learning to learn".

The three levels of rules correspond to the three levels of discourse suggested by Pask (1975). Pask proposes two levels of object language, L^0 for commands and questions used directly in the instruction, L^1 for commands and questions about learning procedures, explanations, or overall strategies, and an observation metalanguage, L^* for describing the system, setting contacts, or specifying goals. Scandura (1977) has discussed the idea of lower and higher order rules where higher order rules are rules which act upon other rules (e.g., generalizations). For Scandura, higher and lower order rules are defined by their respective complexity and context -- a higher order rule could be a lower order rule for another higher order rule. Thus, there seems to be a good theoretical basis upon which to argue that a subject matter/curriculum can be viewed as a set of hierarchical rules and further, that the learning process involves acquiring and organizing rules (i.e., strategies) about how to learn these rules.

What about individual differences? It is suggested that individual differences be considered as differences in rule systems possessed by individuals. While the primary interest is with cognitive and perceptual rules, we can also think of difference in social rules (how to interact with others) and

moral rules (criteria for good and bad) which relate to personality differences. Classical personality traits (e.g., introversion / extroversion, anxiety, authoritarianism, field dependence, etc.) can also be thought of as generalized descriptions of rules for dealing with certain situations or feelings. There have been a number of attempts recently to reconceptualize ability differences in terms of rules (e.g., Hunt, Frost & Lunneborg, 1974; Carroll, 1976) and this work has shown the plausibility of this idea. To the extent that students differ in rule systems, they will tend to organize and structure material differently. Thus, the rules/concepts which comprise the subject matter must be arranged to suit the rule system of the individual.

With this minimal theoretical basis, two major implications for instruction can be identified. The first is that curriculum should be organized so that component concepts or rule are distinct and capable of being reorganized by the learner. This follows from the argument for the constructive nature of learning. Concept and rule learning research suggests that an abundance of examples which illustrate positive, negative and irrelevant attributes are essential to the efficient learning of concepts. Because of individual differences in abilities and personality, different individuals will require examples of differing complexity, difficulty or content. The second implication is that the instructional design should include the facility for utilizing different levels of rules during

the course of instruction. Thus students should be able to alter modes of instruction in relation to different prior knowledge, different learning strategies, or as their level of sophistication within the subject domain changes. Finally, there should be the capability to assist the student with "local" debugging (i.e., specific problems or misunderstanding) and also with "global" debugging of learning tactics and strategies.

The above implications suggest that the learner should have control over the three fundamental aspects which relate to an instructional objective, i.e., content, strategies, and mastery criterion. Thus the student has control over content if it is possible to alter the selection of the concepts and the level of difficulty. Control over the instructional strategies means the capability to alter the sequence of instruction and the presentation modes (e.g., exploratory versus expository). Allowing the student to control the amount of practice, the mastery levels, or access to hints or help constitutes control of instructional standards.

The TICCIT (Timeshared Interactive Computer Controlled Instructional Television) CAI system (see Bunderson, 1974) provides the prototype of a learner control system which allows control over these three major components. TICCIT is organized into 4 levels of instruction, the highest level being concerned with course objectives and student progress,

and the lowest level being the primary instructional components. The subject matter is organized into five distinct files: maps files (objectives and pre-requisites), generality files (concepts or rules), instance files (examples), test files, and "fun" files (options). These different files are accessed via LC. In addition, the student can select an expository or inquisitory mode for examples and can alter the level of difficulty (easy, medium, hard) for either the generalities or instances. Learner control is an integral and necessary part of the TICCIT system. It provides the means whereby the student directs the sequence of concepts or rules and the examples necessary to understand them. It also provides the means for monitoring and managing self-progress. The empirical study to be reported in the next chapter evaluates most of the LC features represented in the TICCIT system. Since the TICCIT system embodies the theoretical ideas developed in this chapter, this evaluation also assesses the theoretical framework.

IV. A Learner Controlled APL Course.

This chapter discusses the design, implementation, and evaluation of a learner controlled CAI course to teach A Programming Language (APL). It should be understood that this study was conducted from an evaluation rather than research perspective. Evaluation studies are characterized by their orientation towards providing sufficient information to make a decision regarding the usefulness of certain methods or media in a particular instructional situation, whereas research (in the traditional sense) has been concerned with testing the validity (i.e., truth) of hypotheses which are intended to be generalizable across all situations (Popham, 1975). Thus, the study was mainly intended to assess how a number of different LC features (and different conditions of access to the LC) would affect the performance and attitudes of the particular groups of students for the purpose of determining the usefulness of LC in this particular instructional context(i.e., learning APL). Of course, it is hoped that the results will also be useful in reaching conclusions about the theoretical aspects of LC and its role in the development of computer-based instruction.

Furthermore, the course was designed, developed and evaluated under "real" instructional conditions with students who genuinely wanted to learn the subject matter (as opposed to "participating for course credit"). Errors

were corrected and poor instructional sequences improved as the course was used and hence the exact content of the course changed over time. This runs counter to the usual experimental paradigm where all "subjects" within the same "treatment" must have exactly the same stimulus conditions. However, from an evaluation perspective, such dynamic conditions are those which would be found in any legitimate instructional context (towards which the conclusions are intended) and hence tend to improve the ecological validity of the study.

This chapter is divided into three sections. The first discusses the nature of APL as a subject matter and in terms of some of the ideas developed in the previous chapter; the second describes the design and implementation of the course APLLC; and the third presents the method and results of the evaluation of APLLC.

About APL.

APL is a very powerful and concise notational system devised by Iverson (1962) which has been implemented as a high level programming language. There are two major aspects of learning APL: learning the details of the notation itself (i.e., the APL symbols and their syntax), and learning basic programming concepts such as variables, data types, algorithms, iteration, branching and so on. Because APL is

fundamentally a mathematical system and some of the primitive functions have specific mathematical content (e.g., trigonometry, boolean or matrix algebra), the mathematical sophistication of the student will play some role in his/her learning rate. Similarly, the student's previous programming knowledge and experience will also influence (sometimes negatively) the learning of APL. Thus programming and mathematical preskills must be taken into account in a study of APL instruction.

An important question in the testing of LC using APL is whether or not there is an optimal hierarchical structure for presenting APL concepts. Obviously, if there is one such structure, allowing the student to alter this via learner control will result in suboptimal instruction. On the basis of the theoretical framework developed in the preceding chapter, it was suggested that no such single optimal structure could exist given the different prior conceptual structures and individual differences of students and therefore the learner should be allowed to reorganize the subject matter themselves.

Is there any such optimal structure for APL? Table 2 lists the chapter contents as they are given in three widely-used APL texts: Gray (1973), Gilman & Rose (1970) and the IBM Independent Study Manual (IBM, 1976). By comparing the columns in the table, it can be seen that there is indeed considerable agreement between these three authors in

the specification of concepts in APL and the order in which they should be learned. Thus, all authors introduce the arithmetic primitives and the concept of variables to begin with followed by the primitives for logical, relational, logarithmic, exponentiation, and maximum / minimum functions. Function definition and modification follow these primitives and then workspaces and system commands. However, apart from this general agreement, there are many differences. For example, reduction, function input / output, matrices, and branching are introduced at different points with different authors. This suggests that as far as the inherent logic of APL is concerned, there are a number of concepts which stand in a more or less hierarchical relationship (namely, a subset of primitives, variables, function definition, system commands) and other concepts which can be related to these in various ways (and hence are not part of a strict hierarchy).

Further evidence on this point was collected as part of this thesis. O'Neal' (1977) explored learner control in a task involving the learning of a subset of APL concepts. O'Neal arranged the task structure so that this subset of concepts formed a strict hierarchy as shown in Figure 2. A number of individuals very familiar with APL were given an alphabetic list of 11 of the concepts O'Neal used ("stuffing" and "counting" were left out due to ambiguity) and asked to arrange them in the best teaching structure. The results of this exercise are given in Figure 3. It

appears that for this particular subset of APL concepts, the hierarchicial structure as perceived by these subject matter experts is weak.

The preceding discussion of organizational structure for teaching APL can be summarized by saying that there are a number of basic concepts with strong interdependencies among them (and which should be organized in a particular hierarchical sequence) and a set of weakly dependent concepts which can be related to these basic concepts in various ways. These basic concepts are variables and assignment, data structures, primitives, defined functions, and system commands (workspaces). It is suggested that a spirally organized curriculum which involves successive elaboration of the basic concepts in 3 or 4 levels would be the best way to organize APL teaching. This suggestion is based upon the theoretical ideas discussed in the preceding chapter and the organization reflected in major APL texts.

Table 3 shows the concepts for such a spiral curriculum with 3 levels. Level I introduces variables and assignment for numbers, numerical and relational primitives, combining primitives into algorithms, and error messages. Only numerical scalars and vectors are introduced in this first level. Level II introduces literals and matrices, more primitives, the operators sum and scan, defined functions, system commands, and errors associated with defined functions and system commands. Level III involves the

introduction of the remaining primitives, inner and outer products, further aspects of defined functions, system variables, and public libraries. Note that at each successive level either a basic concept is elaborated (e.g., primitives with scalars, vectors, primitives with matrices) or a new concept is introduced which depends upon a previous one (e.g., defined functions depend upon primitives). This structure also has the property that each level consititutes a coherent unit; a student could quit after each level and understand something useful about APL.

The study of actual APL usage and errors made in APL programming emphasizes the human performance capabilities or limitations in the use of APL in contrast to the rational analysis of its logical structure. In effect, such information reveals what people actually learn (and fail to learn). This type of analysis should play as much a role in the design of instruction as rational task analysis in terms of indicating necessary and unnecessary preskills or appropriate levels of difficulty.

A study by Saal & Weiss (1977) provides descriptive data on APL usage. Their sample consisted of 32 IBM application workspaces ranging over areas such as plotting, text editing, linear programming, statistics, and digital circuit design. Their findings can be summarized by the following points:

1) User defined functions tend to be short with the median

length being 5 lines and 20 percent of all functions being "1 liners".

2) Internal documentation of code was rare (about 1 percent of all lines).

3) Ambiguity due to local / global confusion was relatively rare.

4) Recursion was used in 6 of 32 workspaces (which may be a sampling artifact).

5) 90 percent of all APL operations consist of primitives and subscripting, and the use of operators (reduction, scan, inner & outer products, catenation) account for only 4 percent combined.

6) The seven primitives 'plus', 'and', 'or', 'floor', 'ceiling', 'times', and 'minus', account for 95 percent of all reductions while the three primitives, 'plus', 'and', 'or',, account for 75 percent.

7) In terms of subscripted objects, 56 percent were vectors, 43 percent were matrices and only 1 percent involved 3 dimensional arrays.

8) The combinations of 'plus dot times' and 'and dot equals' account for over 70 percent of all inner products.

9) Three primitives, 'branch', 'row' ('reshape'), and 'ravel' ('catenate'), account for 75 percent of all monadic use in APL and the seven primitives: 'assignment', 'plus', 'compression', 'times', 'equals', 'rank' / 'reshape', account for 75 percent of all dyadic

usage.

This last point leads to the "80-20" rule, namely that 80 percent of the observed usage is accounted for by 20 percent of the available functions. Saal & Weiss conclude that many features of APL are used extremely infrequently and hence could be eliminated from introductory teaching of APL.

There are three levels of errors which can be made in APL programming: (i) function execution errors, (ii) workspace management errors, and (iii) logic errors in defined functions. Errors of type (i) give rise to one of the eight APL error messages and an indication of where in the expression the error occurs. Errors of type (ii) arise from the use of system commands and variables and also from exceeding workspace limitations. Type (iii) errors involve errors in function iteration, branching, subroutines or input/output and may or may not result in explicit error messages of type (i).

Table 4 provides some data on the frequency of occurrence of type (i) and (ii) errors in student programming collected as part of the present study. The data is based upon a sample of 148 partial or complete listings of terminal sessions[9] collected from the recycling bins of a

---

9. Of the 148 listings, 52 were of complete sessions. The average duration of these sessions was 31.4 minutes using 2.6 seconds of CPU time.

terminal room over a one month period. The sample represents a cross-section of novice and experienced APL programmers across different application areas; the error frequencies would undoubtedly differ in nature with programming experience and the nature of the application. As Table 4 shows, function execution errors account for over 91% of the total errors in these sessions with value and syntax errors accounting for over 50% of the execution errors. Workspace management errors account for only about 8% of the total errors and about 50% of these are incorrect command errors. The value errors stem mostly from unassigned variables; however some arose from syntactic mistakes such as missing a space between a function header and an argument (e.g., MEANX for MEAN X). Syntax errors arise when a function is missing its proper argument(s), e.g., when a dyadic function has only one argument. Syntax errors can also arise from indexed variables where the variable name is missing. As for incorrect command errors, the majority of these were due to incorrectly spelled workspace, function, or variable names in system commands.

The implications of this data for the design of APL instruction are twofold. First, given a knowledge of the frequency of certain errors, additional emphasis or attention should be placed on teaching aspects which are highly error prone as well as diagnostic testing to locate missing pre-requisite skills or tasks. For example, many errors arise because students do not understand how to use

arguments in defined functions. Sometimes students do not understand the relationship between values used in the header statement and the body of the function -- this usually leads to VALUE errors. Figure 4 shows an example of SYNTAX errors arising from a misunderstanding of function arguments. This example also illustrates the multiple causality which can occur in the generation of error messages. Error data suggests that the concept of function arguments, particularly in defined functions, requires extra attention.

The second implication of information about errors in APL is that instruction is needed to teach the student how to correct them (i.e., debugging). Each of the three error types will involve different debugging strategies (rules). Type (i) requires the general understanding of what each error message means and how to use the location information to isolate the error. Given that execution errors account for the major proportion of errors, this clearly should receive the most emphasis. Type (ii) errors require checking the names of workspaces, functions, or variables (i.e., via)FNS,)VARS, etc.) or altering the workspace parameters (i.e.,)WSSIZE,)STACK,)ERASE, etc.). Type (iii) errors require the teaching of techniques such as checkpoints or tracing in order to isolate logic bugs. The greater the detailed understanding of these three types of errors, the better the debugging rules taught to the student.

The above discussion suggests three levels of rules which could be involved in the learning of APL:

Level 1: Concepts and procedures of APL and their inter-relationships. These are the rules inherent in the explanation of concepts and principles (expository mode) and examples (inductive mode).

Level 2: Procedures for what and how to teach or learn APL. This would include a strategy such as the spiral organization of concepts. It would also include the knowledge of APL usage or errors to emphasize certain distinctions. This includes minor confusions such as between the negative sign and the minus operator or more major misunderstandings such as correct function arguments or the difference between scalars, vectors and matrices. In some cases, strategies can be taught to help minimize errors,e.g. debugging strategies.

Level 3: Rules about how to learn a programming language. This would include the necessity of having both tutorial and practice sessions, how to evaluate mastery of concepts (through problem solving), and how to modify the subject matter to suit personal styles.

These three levels of rules should be accounted for in the design of an APL course. Moreover, the student should have a means of controlling each level of rules. This includes the capability to control the selection, sequencing, mode of presentation, level of difficulty or amount of practice in the context of APL.

The design and implementation of APLLC

APLLC was written in the COURSEWRITER II language and implemented on the IBM 1500 system operated by the Division of Educational Research Services at the University of Alberta. The subject matter of the course is organized into two separate files; concepts and examples. The course consists of 7 major topics which each comprise about 3-7 concepts. The concepts are composed of 5-10 instructional units. The topics and concepts are given in Appendix I.

The LC options were implemented via a "CONTROL" frame feature. The CONTROL display is shown in Figure 5. It provides six major options; review the present concept; branch to the next default concept; branch to the examples for the present concept; branch to the topic index; make a comment; or change the difficulty level (easier/harder) of the presentation. The selection of examples provides a display of the available examples for that concept (range 2-5). The selection of the current concept index allows the student to branch to any other concept in the current topic. In addition, the concept index allows access to the overall topic index so that the student can select another topic. The comment option in the CONTROL frame allows the student to make comments to the course author. The master topic index allows selection of two further options; a glossary and student progress records. The latter provides the student with information about the concepts/ topics which

have or have not been covered, the last concept studied, and a summary of performance.

The difficulty level option caused the presentation of simpler or more difficult explanations of concepts when they were available. Sometimes, there would be only an easier or harder alternative explanation or only one level. The distinction between the easy and difficult levels involved the degree of conciseness, redundancy, use of analogies, and alternative instructional strategies. For example, the easy example for introducing defined functions used a niladic function which produced a box; the default level a monadic function for computing the mean of N; and the hard level involved a dyadic function for text editing. The easy level concentrated on the idea of sequential execution without the complication of arguments whereas the default and hard levels introduced arguments at the same time.

The self-reported level of mathematical and programming knowledge was also used to select instructional sequences. However, unlike the difficulty level, the selection process was automatic and not under the learner's control. Thus an individual who has no programming experience would get much more elementary and complete explanations of the concepts. Similarly, for those who indicated no knowledge of certain mathematical areas, the corresponding primitives were bypassed (e.g., no knowledge of trigonometry would cause bypassing the circular primitives). In some cases, there was

some overlap between the selection of material which was easier/harder and selection for the mathematical or programming level. For example, the explanation of matrices as tables is an easier mode than one used for those with no knowledge of matrices.

The topic index and each concept index contained a further feature -- the RETURN option. This feature was based upon a push-down, pop-up stack of label locations which allowed the student to return to the concept interrupted by using LC. Each time the CONTROL display was accessed, the present concept was loaded into the stack. The RETURN feature allowed the student to interrupt a concept, move to another concept, and then automatically return to the interrupted location. An unfortunate limitation of the particular implementation of the RETURN feature was that the stack had a capacity of 15 labels per terminal session and overflow resulted in overwriting of the stack.

Testing in the course was of two types: embedded questions and unit post-tests. The embedded questions were one to three questions which followed the presentation of each new concept. Their purpose was to ensure that the student understood the immediately preceding instructional sequence. Failure to answer the questions correctly resulted in either a repeat of the sequence or remedial sequence to correct a specific misunderstanding indicated in the answer(s) to the question(s). Post-tests came at the end of

every concept in the course (e.g., numerical primitives, assignment, error messages, etc.) and usually consisted of 10 questions covering the entire concept. Learner control was always turned off during the unit tests to prevent the student from skipping them. If the student got less than 50 percent of the post-test questions correct, the concept was repeated. If the score was above the 80 percent level, the student was branched to the next concept. If the score was in the 50 - 80 percent range, the student was given the option of reviewing the concept or going on to the next one.

The strategy is a built-in compromise between giving the student full learner control over the amount of practice and degree of mastery achieved and allowing no control. Because APLLC was designed to fill a genuine instructional role, it was felt that a minimum level of instruction had to be guaranteed if the course was to be instructionally effective. On the other hand, in keeping with the purpose of allowing the student to control all instructional parameters, some control of practice and mastery level was necessary. It is felt that the above strategy allowed control over evaluation in the crucial range of decision for a learner. It should be noted that the design of APLLC did not allow a complete assessment of LC with respect to allowing students to define the amount of practice and mastery levels.

To summarize, a course on APL was implemented with the

following LC features (see Figure 5 and Appendix I):

* the capability to select either a concept or examples of a concept for study (next, example)
* the capability to review a concept or the associated examples (repeat)
* the capability to select any concept in a particular topic (concept index)
* the capability to select any topic in the course (topic index)
* the capability to use a glossary, study course performance or progress or make comments to the course author
* the capability to alter the level of difficulty of the instructional presentation (easier, harder)
* the capability to improve the level of mastery when the achievement level on a unit post-test falls in the 50 - 80 percent range

APLLC was implemented in the Fall of 1976 and used by about 60 high school students who belonged to computer science and math clubs. The LC features were optional and could be accessed via a special keystroke at the end of every complete screen display (except during post-tests). The students received the topics in a default sequence (data structures, primitives, combinations, defined functions, system commands & variables, errors) unless they used the learner control to alter the sequence or mode.

During this time the course was debugged technically and instructionally. On the basis of performance records, student comments, and comments of other APL teachers and experts who went through the course, the course content was modified considerably-- terminology was changed, more embedded questions were added, and the students were asked to type APL expressions for answers as often as possible. The most major change was the sequencing and organization of the concepts into the spiral structure described in the previous section. At the end of the field testing, the course was considered instructionally sound and ready to be evaluated with respect to the utility of learner control. Thus, the revised form of the course used in the evaluation was not tested out beforehand; furthermore, it underwent modifications as students began to use it.

Evaluation of LC.

The concern of the evaluation phase of this research is to assess questions such as:

1. What are the effects of differing degrees of LC on performance and attitudinal outcomes?

2. Which performance or affective measures are sensitive to LC effects (if any)?

3. What individual difference variables predict and reflect the use of LC?

4. Can both facilitative and inhibitory effects be

demonstrated for LC for the same individual in different parts of the course or for different options?

5. What relative proportion of students show some degree of performance and attitudinal benefit from LC?; what proportion show no effect?; what proportion show negative effects?

6. How does the use of LC change with the sophistication of the student both in terms of differences in pre-entry skills and knowledge as well as learning which occurs during the course?

7. If LC results in a slower rate of learning, does this reflect a "deeper" mastery or comprehension of the subject matter (i.e., greater retention of the material) or lost time due to inefficient course management?

In order to address these questions, APLLC was modified to allow for three possible conditions. The first condition (no-LC) allowed no LC options whatsoever (save for the capability to make online comments to the instructor or author). In this condition, the student received the default sequence of concepts according to the spiral organization and a pre-specified example for each concept. The second condition, learner control/optional (LC/O) had the same default sequence and example presentation; however, access to the learner control options was available at the end of every complete screen display via a special keystroke. In the third condition, learner control/full (LC/F), the student was returned to the CONTROL display at the end of

every concept or example. The selection of the next concept or example was up to the student. If the student selected the next concept option, he / she would get the default sequence as in the no-LC condition. Thus, these three conditions provided for no, optional, or full access to the learner control features available.

An explanation of the LC features was given to the LC/F and LC/O groups. As part of the explanation, students were shown the effects of the various LC options and required to demonstrate that they could use these options. There was not, however, extensive practice in the use of the options.

Students from two different groups were randomly assigned to one of these three conditions. One group consisted of high school students (N=20) who were learning APL as part of a math club activity. The second group consisted of graduate students (N=18) who were learning APL to be used in conjunction with a course in multivariate statistics. The university group consisted of about equal males and females while the high school students consisted mostly of males. The high school students took the course on their own initiative (attendance was voluntary) and were probably intrinsically motivated while the university students took the course as part of their class activity and hence may have had more of an extrinsic motivation. This difference between the two groups in motivation is a conjecture, however, and was not measured in any direct way

in the study. Furthermore, the high school students were learning APL as a programming language, whereas the university students were learning it as a tool for learning statistics. The differences between these two groups in age and motivations could be expected to produce differences in their learning progress and use of LC. In particular, it could be expected that the high school students would complete the course in less time and also would use the LC features to a greater extent due to their greater interest in the subject matter.

The performance measures included the total number of correct responses from the post-tests for each topic, the total time on each topic, and the ratio of total correct / total time. These measures reflect the instructional effectiveness of the material and also the effects of altering the sequence, mode or mastery criteria via the learner control. Students who select extra practice (via examples or reviews) will answer more questions and hence have the possibility of having a larger total correct. Similarly the total time can be increased through the use of the learner control features. The correct / time ratio is a measure of learning rate.

Two independent measures of learning were made with the university group. When most of the students were about halfway through the course, they were asked in a class period to write a function to compute the average of a

vector. The second measure consisted of a question on their final exam which involved writing an APL function to compute multiple regression weights and multiple correlation coefficents. These two scores were intended to gauge the degree of transfer of learning in the use of APL.

The attitudinal measure consisted of a 10 item questionnaire which was given near the end of the course conditional on the student's total time being at least 3 hours. The scale was a 5 point Likert type scale with "Completely Agree" at one end and "Completely Disagree" at the other end. The 10 items on the questionnaire are given in Table 5. The items were intended to gauge various affective reactions to LC and the course. Two of the items ask specifically about the students feelings of control (6, 10) while items 1,2,4,7,9 attempt to gauge the learners degree of frustration, perception of course organization, boredom, fulfilment of expectations, and judgment about the completeness of the coverage. Items 3,5,8 measure general reactions to CAI. Half of the items were negatively stated and half were positively stated.

Three covariate measures were taken; two were concerned with preskills and the third with a personality dimension. Students were asked to rate their degree of programming experience as none, some, or experienced and their mathematical familiarity with 7 mathematical areas: trigonometry, factorials, logarithms, matrix algebra,

modular arithmetic, boolean algebra, and calculus as either nil, some, or lots.[10] The third covariate measure was the learners score on an internal / external locus of control score. The scale consisted of 23 paired items taken from Rotter's (1966) scale with the filler items omitted to abbreviate the test. Rotter's scale has been found to be a reliable measure of an individual's belief in the personal control they exert over their life as well as having construct validity with other related tests (Rotter, 1966). Individuals with a high score on the scale manifest a strong belief in the external control of their life by factors such as chance, luck or fate. Individuals with a low score manifest a strong belief that their experiences are the result of their own actions and behavior. Research has shown that internal / external locus of control, as measured by this scale, relates to many other personality characteristics which could be important in instruction, e.g., achievement motivation (McGee & Crandall, 1968) or expression of humor (Lefcourt, Sordoni & Sordoni, 1974) which affects the student's motivation during learning. Of all the personality measures which could be made, this one would seem to have the most relevance to the use of learner

---

10. In addition to providing a measure of mathematical level, these scores were used to bypass topics the student had no knowledge of (e.g., circular primitives for trigonometric functions) or provide remedial instruction for others (e.g., matrix algebra, logical primitives). No reliability or validity data was collected for this self rating measure.

control. It should be noted that the ommission of the filler items to abbreviate the questionnaire may have affected the reliability and validity of the scale and hence any comparisons between the results obtained using this abbreviated form and research using the full form must be very tentative.

To summarize, two different groups of students (high school and university) were assigned to one of three LC conditions: no learner control (no-LC), optional LC (LC/O), and full LC (LC/F). Dependent measures were total correct, total time, correct / time per topic and attitudes at the end of the course. Mathematical and programming knowledge and internal / external locus of control score were the concomitant variables.

<u>Results of the Evaluation of APLLC</u>

First we will consider the use of the learner control features for the students who had learner control capability (both LC/O and LC/F). The total use and percentage of use for the LC features is given in Table 6. Looking at the CONTROL options, the "next" and "index" features account for the largest proportion of LC use. The difficulty levels (easy/hard) constituted 25.1 percent of the use. The example, repeat, and comment features were used relatively

infrequently. The LC use in the Topic index indicates that the Progress Report features were heavily used and that students used the Topic index to select another topic about 15 percent of all LC requests. Since only 15 percent of the 256 total LC selections from the CONTROL frame were for topic changes, we can conclude that only 4% of all LC selections were for change of topic overall and that 79 percent of accesses to the concept index were to select another concept within the same topic. As far as the progress report options are concerned, the options of concepts done, concepts not done and performance summary were about equally popular.

The use of LC options via the CONTROL frame combined across both student groups is shown in Figure 6. As the graph shows, there were differences between the total number of accesses between the LC/O and LC/F for the next, index, and difficulty level features. The LC/F group had a higher total number of accesses for the next and difficulty level features and fewer index selections than the LC/O group. In terms of change over time, the average number of accesses between the two groups for each topic in the default sequence is given in Figure 7. This figure shows that there is a considerable difference in use of LC to begin with, but that the difference essentially disappears by the fourth topic. A one way analysis of variance with the topics as repeated measures indicates that the differences between topics is significant at p=.01 (F [6,60] =5.8). Detailed

analysis of the use of the CONTROL features over topics indicates that the number of accesses to the CONTROL features was greatest for the LC/F group for all features, except for the index feature which was most used by the LC/F group to begin with, but more heavily used by the LC/O group later.

These results suggest that the capability to move to another concept in the same topic is important to the student while the capability to move to another topic is much less so. The capability to alter the difficulty level also appears to be an important LC feature to the student. On the other hand, the examples were not well used -- perhaps because the illustrative examples in the concept explanations were sufficient.

As far as the difference between the optional and full LC groups is concerned, it is unclear why the LC/O group used the index option more often or why the LC/F group changed the difficulty levels more often (Figure 6). The trend for the two groups to come together in their amount of LC use as the course progressed may represent an optimal or asymtopic level of LC use that both groups reached. The increase in the index feature by the LC/O group over time may reflect the effects of increased practice with the LC. This data suggests that there are differences in the effects of LC over time, or with practice. In general, the average number of accesses to the different features was

proportional across all topics (declining overall), suggesting that no differential change in the use of particular LC features.

The average number of accesses to the CONTROL display for each topic for the two student groups and LC conditions are given in Table 7. As can be seen, the high school students in the LC/F condition used the LC options more often than those in the LC/O condition in almost all the topics. For the university students, however, the LC/O condition had more average accesses to the control frame in the first 3 topics but less than the LC/F condition in the last 4 topics. Thus, the high school students used the LC more than the university students under the full access condition whereas the university students used it more under the optional access conditions (at least initially). This data suggests that there were differences in the way the LC was utilized between the two student groups and this may have been due to motivational differences, however, these differences were not statistically significant.

Looking now at the effects of LC on the three performance measures, consider first the effects of no-LC, LC/O and LC/F on average total completion times. As Figure 8 shows, for both high school and university students, the groups with LC took longer to complete the material. However, for the university students, the LC/O condition had the longest times whereas for the high school students, it

was the LC/F condition. The average total completion times by topic are shown in Figure 9a (university students) and Figure 9b (high school students). In the case of the university students, the no-LC group takes less time in every topic than the two LC groups except for topic 4 where the no-LC group took longer than the LC-F, but slightly less than LC/O. In the case of high school students, there appears to be a reversal of the times for the 3 groups between the first 3 topics and the last 4. In the first part of the course, the no-LC group takes the greatest average time with the LC/O taking the least; in the last part of the course, the LC/O group takes the greatest average time with the no-LC condition taking the most time. When these times are averaged, the no-LC group takes less time than the two LC conditions. These results suggest that in general, LC options cause the student to take longer in completing the material. This time is probably spent in the organization of learning rather than in learning itself. The present study does not allow the separation of these two aspects of time utilization.

Comparison of Figure 9a and 9b (as well as Table 8) indicates that the high school students took less time overall to complete the course. In terms of difficulty level and pre-skills, it would be expected that the high school students would take longer, not less time. The most likely explanation of this result is that the university group were motivated to learn APL more thoroughly than the high school

students because they needed it to complete course work. The high school students, on the other hand, were taking it "for fun" and were perhaps less diligent. Unfortunately, no independent measures of performance were made with the high school students so confirmation of this hypothesis from the present data is not possible.

Figure 10 shows the average number correct across all post-tests for students in each of the three LC conditions. It is clear that students in the no-LC group score more correct that the LC groups and that the LC/O condition results in more correct than the LC/F condition. This suggests that with the greater degree of LC allowed, the student is more likely to make a larger number of errors. This is in spite of the fact that LC allows more practice, review, and more examples. Table 8 shows the average number correct across each topic for each LC condition. Note that the cells in Table 8 do not represent the same N because not all the students in each group completed every topic. Also note that the average total correct (last column) is the average of the total correct scores of all the student in that group and was calculated separate from the scores for each topic. It can be seen from the Table that the differences between the scores are essentially the same across the three LC groups (except for LC/F in topic 3) suggesting the overall difference in number correct was not related to particular topics.

Figure 11 shows the ratio of the average number correct to average total time across all subjects for the three LC conditions. This ratio is a measure of learning rate, i.e., the number correct per time spent learning. For both student groups, the ratio is higher for the no-LC conditions than the LC conditions suggesting that the no-LC condition resulted in a higher learning rate. For the university students, the LC/F group had lowest learning rate while for the high school students, the LC/O had the lowest rate. This difference between the two student groups reflects the differences in total times discussed above.

To summarize the results of the performance measures, it seems that in general the amount of time required increases with the amount of LC available and the number of correct and learning rate decreases with the amount of LC available. In other words, increased use of learner control appears to cause the learner to take longer, score fewer correct and achieve a lower learning rate. The analysis of variance of the three performance measures which is shown in Table 9 indicates that none of the effects described above is strong enough to be significant at p=.05 except for the differences between topics.

What about the effects of LC on attitudes? The average scores for the 10 items in the attitude survey (see Table 5) for the two student groups and three LC groups is given in

Table 10.[11] "Completely agree" was assigned assigned a value of 1 while "Completely disagree" was a value of 5. The composite score represents the sum of the scores on the 10 items with the negative items transformed to the positive direction (i.e., they were subtracted from 5). It provides a measure of overall positive attitude towards LC and CAI. Analysis of variance on the items between the three LC conditions and the two student groups indicates that there were no significant differences. Thus, there does not appear to be any general attitude differences between the three LC groups for the items as a whole and the composite score is essentially the same for the three conditions. The responses of the university students in the LC/O and LC/F conditions relative to the no-LC condition tend to be in the same direction with the exception of item 2. Thus the university students in both LC conditions indicated that they felt less frustrated (item 1), less bored (item 4), agreed that CAI had disadvantages (item 5), felt that they did not need more LC (item 6), that they learned as much as they expected (item 7), that they would like to have all their instruction via CAI (item 8), and that CAI did not allow as much freedom as they wished (item 10), in comparison with the no-LC condition. The high school students in both LC conditions indicated that they found the material well organized, felt

11. Because of problems in the recording of data and students not completing the segment in which the questionnaire was located, attitude data was available for only about half of the students in the two groups.

that they would have learned more via traditional classroom instruction, that CAI has disadvantages, that more student control was needed, and that they would not want to take all of their instruction via CAI. On the remaining items, the LC/F and LC/O groups went different ways in relation to the no-LC group.

Items 6 and 10 specifically asked about the control of instruction. For item 6 ("the course should allow more student control and selection"), we see that the LC groups reacted in different directions relative to the no-LC group. The high school students in both LC/O and LC/F felt that more LC was not needed relative to the no-LC group; the score was the same for both LC conditions. Both university LC groups ageed that more LC was needed, however, the LC/F group agreed more than the LC/O group. For item 10, ("CAI provides a lot of freedom in learning"), the LC/F group disagreed with this statement more than the LC/O group for both high school and university groups. In general, the attitude data shows no systematic differences between the LC conditions.

In order to identify any significant effects of the covariate measures on the attitudes, the mean score for each measure was used to divide the students into high and low scoring groups on the measure. For programming preskills, this divided the groups into those with no programming experience and those with some experience. The average

scores for the high and low programming skills groups across both student groups for the 10 items and composite score is given in the top of Table 11. In general, the low programming skill group disagreed more with all items than the high skill group except for item 5 ("CAI has lots of disadvantages") and item 6 ("should be more learner control"),i.e., they agreed that CAI has lots of disadvantages and more learner control was needed. Hotellings T test on the 10 items and the composite measure indicated that there were no significant differences between high and low groups.

The middle portion of Table 11 shows the scores for the high and low mathematical skills groups. Only items 5 and 8 were significantly different (t-tests, p=.05, 2-tailed). The low math group disagreed more with the statement that learning via computers has lots of disadvantages and agreed more with the statement that they would like to take all of their instruction via CAI. Thus, the low math skills group expressed more favorable attitudes towards CAI on items 5 and 8.

The bottom of Table 11 gives the scores for the high and low score groups on Rotters' scale. The high scores indicate external locus of control, the low scores indicate internal locus of control. There was no difference between the composite scores of the high and low group and no systematic differences between the item scores for the

groups. The low (internal) group agreed more with item 6 that there was not enough learner control than the high (external) group. The low group also felt that they had learned less than they expected in comparison with the high group. These two responses are in line with expectations about internal controlled students preferring more control over learning and not achieving their own expectations. However, in general there was no significant difference (Hotellings t, F,10,10=0.93, n.s.) between the degree of internal/external locus of control as measured by Rotter's scale or the attitude items for the high and low groups.

The final results to be presented pertain to the 2 programming quizzes given to the university group as measures of independent learning. The first quiz (midterm) asked students to write a function to compute the average of a vector; the second quiz (final exam) asked students to write a function to compute regression coefficients. Both quizzes were scored out of 5. Table 12 gives the means and standard deviations of the scores for the three LC conditions. As can be seen from the Table, the scores for both quizzes were the highest for the students who did not have learner control. The students who had LC did not show a higher degree of transfer in an independent measure of learning. Table 13 displays the correlations of the scores on the two quizzes with 3 performance measures and the composite attitude measure from APLLC. The data indicates that higher scores on the quizzes were associated with less

total time and a higher learning rate in APLLC as well as a somewhat negative attitude towards CAI/LC. The negative correlation between the quiz scores and the total time on APLLC could be due to the fact that the students who already knew some APL would have spent less time on APLLC but done better in the quizzes.

In addition to these quantitative results, a few qualitative observations are pertinent. From casual conversations with students, and their comments made via the comment feature, it was clear that for some students, LC was valuable and for others it was confusing. The former group of students seemed to be able to distinguish the content of the course from the instructional logic and as a consequence were able to make appropriate use of the LC features. Students who could not do this tended to be confused about the use of LC. It also seemed to be important that the student grasp the overall structure of the course (in terms of topics and concepts) in order to use the LC; many students who became confused using the LC seemed to lack this knowledge of the topics and concepts. An elaborate map system as used in TICCIT would be helpful for these students. There were also a number of students who were frustrated because they did not have enough LC; they requested further features such as backup one frame, help/hint mode, and alternative explanations.

V. Conclusions and Implications of the Study

The results of this study need to be considered:

1) in terms of the value or effects of LC in the specific context of a computer-based APL course,

2) in terms of the relationships between LC and the design of CAI systems and,

3) in terms of the theoretical framework for LC developed in Chapter 3.

Each of these considerations leads to conclusions about learner control at different levels of discourse and implications for different aspects of instructional psychology.

The value of LC in APLLC.

The first consideration involves conclusions about the best LC situation for a CAI APL course. The results given in the previous chapter suggest that as far as the effects of LC on performance are concerned, increasing the degree of LC access tends to reduce performance in terms of greater total time required, fewer average correct, and a lower learning ratio as well as producing less learning transfer. In terms of the attitude measures, there was no strong effect of the LC conditions on affect, either negative or positive. Nor do differences in mathematical or programming preskills or differences in internal/external locus of control seem to have any major effect on the use of LC or attitudes towards

to LC. Taken together, these results suggest that in this particular instructional context, LC does not seem have any beneficial effects on performance and attitude measures.

On the other hand, some of the LC features were heavily used, suggesting that they were indeed important to certain students. The capability to branch to other concepts within the same topic, to alter the difficulty level, and to examine progress were popular options. Thus, even though the results suggest that LC is not beneficial to learning progress in general, it is clear that the LC features were used considerably by some students (and hence were useful in some way to these students). Qualitative observations during student sessions also support this conclusion. Some students used LC to review and cross-check new material with that already learned.[12] Given these circumstances, it seems best to compromise and allow the LC features to be used optionally, but provide a default sequence and structure to maximize performance. It would not be a good idea to provide students with full LC, at least as far as the present type of LC is concerned.

Because this study was designed specifically as an evaluation study, the above conclusion regarding LC is only relevant to the particular instructional context

---

12. This suggests a major limitation of the design of the study, i.e., the lack of a within individuals comparison for the effects of learner control.

investigated here (i.e. the particular student groups and subject matter). Most of the students taking APLLC (in both student groups) did not know any APL beforehand and most had no programming experience. Students who already knew some APL or were experienced programmers may have produced different results with the LC since they could be expected to already possess a conceptual structure of APL to build upon. Even though a (self) rating was made of the students mathematical and programming experience, this single quantitative measure is likely too crude to be useful. What is really needed is a qualitative measure of the student's conceptual network in terms of programming or mathematical knowledge.

The course was tutorial in nature and this probably affected the value and usage of LC. The tutorial mode of instruction provides a fairly strong instructional structure (i.e., presentation, question/problem, response, response analysis, feedback, branching). A subject matter or task in which the concepts and procedures are less hierarchical in nature (e.g., psychology, english, geography, etc.) may benefit more significantly from LC. It seems likely that a problem solving or inquiry strategy would have lead to different effects since the learner control would have been more crucial to these types of strategies. Finally, because of the inherent hierarchical nature of APL, it may not be a subject matter which is enhanced by LC.

The results also cover only one particular way of implementing LC. Alternative methods of providing LC might produce different results. For example, in the TICCIT system, the LC options are implemented as special keys on the keyboard. This may facilitate the use of LC by making it more physically accessible. A more general way of implementing LC would be to provide the student with a control language which allows the specification of instructional parameters in a form such as:

SHOW HARD(ER) EXAMPLE

BACKUP ONE CONCEPT

PROVIDE ALTERNATE EXPLANATION

SKIP CONCEPT

SUGGEST STRATEGY (HINT/HELP)

Such a control language would allow the student to have a finer degree of control over the instructional variables. Of course whether such a generalized LC is beneficial to learning still remains to be assessed.

A final consideration is that although students were required to demonstrate their understanding of the LC options, they did not have an opportunity to explore the consequences of trying out different learning strategies. Thus, it is possible to interpret the results of this study in terms of the effects of different degrees of learning how to use LC rather the actual effects of using LC to learn. Because students have had little experience with a learner control mode of learning in traditional instruction, a major

problem with any LC study is ensuring that the results are not due mainly to the amount of practice achieved with the LC.

LC and the design of CAI systems.

These remarks bring us to a discussion of the second major consideration, namely, how LC relates to the development of CAI. The introductory chapter discussed how the perspectives on the role of student control in learning differs between CAI and CAL advocates. The type of LC investigated in this study is within the tradition of frame-oriented CAI in which lessons are constructed on a frame by frame basis. However, there are a few "intelligent" CAI systems [13] such as the Basic Instructional Project (Barr et al., 1976) or SOPHIE (Brown & Burton, 1975) and SCHOLAR (Collins et al., 1975) in which LC is an integral part of the instructional process. BIP is a programming laboratory which consists of a curriculum network that is mapped onto the programming skills mastered by the student. The system

13. The term "intelligent" CAI refers to the idea that the system iteif should possess an understanding of what the student is doing and the effects of certain instruction actions on student learning. Examples of "intelligent" capabilities would be the ability to compare two paraphrases of the same idea and correctly identify them as the same, to be able to deduce or infer why a student has answered incorrectly and them construct a teaching sequence to correct the misunderstanding, or to check for inadequacies or inaccuracies in the subject matter provided.

automatically selects problems which involve skills the student is ready to learn. The student directs the actual learning process by commands which control instruction, the BASIC interpreter, or elicits information and problem solving/ debugging aids. Thus, BIP involves a highly organized task domain with well defined concept inter-relationships but the student fully controls the major parameters of learning.

SCHOLAR and SOPHIE are prototype socratic systems which engage the student in an instructional dialogue in natural language. The system incorporates heuristic strategies for teaching certain concepts and helps students test their understanding of the subject matter. However, the student controls the course of learning via the specific information elicited and questions asked. The major distinction between traditional frame-oriented CAI and these "intelligent" CAI systems is that the latter are organized as knowledge networks which represent an understanding of a subject matter. By providing the student with a high level interaction capability (either a powerful command language or natural language), the student can learn from this network in any manner desired with intervention when the student needs help or guidance. Adding LC to traditional CAI (as in the present study) is really an attempt to move frame-oriented organization towards an "intelligent" network organization.

However for LC to really become an integral and significant part of computer-based instruction, it will be necessary to develop methods of qualitatively representing what the student presently knows about a subject (i.e., the student's conceptual structure) in terms of the detailed facts, concepts, procedures, etc. When such a model of the student's knowledge exists, it will then be possible to compare the task/subject matter structure with the conceptual structure of the student and determine exactly what skills the student has already mastered and what is needed next. This would permit the system to determine <u>what kind</u> and <u>how much</u> LC is appropriate to optimize the learning environment. The task selection algorithm of BIP and the "student models" developed by Self (1974) are steps in the direction of qualitative representation of student progress.

<u>Theoretical Implications.</u>

The third and final consideration relates to the theoretical framework for LC. The data suggest that the capability to branch to concepts within the same topic is more important than the capability to branch to other topics, i.e., that the more important type of sequence control is local rather than global. In other words, in developing a conceptual structure of APL, it was more

important to be able to link concepts closely related (i.e., at the same level of discourse) than those more hierarchically related (at different strata). In terms of mode, students did not make much use of examples suggesting that an inductive mode was not important and that exposition of concepts was sufficient. The heavy use of the progress features suggests that self-evaluation of progress was important in terms of what had been covered and performance level. This provides evidence that self-monitoring and the formulation of rules about how to learn a subject matter was attempted.

The failure of the measures of knowledge level in mathematics and programming and inner/external locus of control to affect performance or attitude measures is puzzling since these characteristics should make a difference to the learning of APL and the use of LC. It is possible that the self-report of mathematical and programming knowledge did not produce accurate or detailed enough measurement. It is also possible that the differences in the two groups studied were not very great and hence not strong enough to produce any effects. It could also be that the particular tutorial structure of the program was too narrow to allow such differences to manifest any significant effects. It is well known that individual differences tend to be minimized when the task or situational demands are strong.

The failure of the LC groups to score higher in the programming quizzes casts doubt on the idea that decreases in performance during learning are due to a greater organizational effort which will result in better permanant learning and retention. In terms of the theoretical ideas developed in Chapter 3, allowing greater freedom to the student in the reconstruction of the subject matter should facilitate learning. However, it is also possible that this freedom can interfere with learning if an optimal strategy is already inherent in the subject matter for the majority of students. It is possible that the organization of APLLC represented an optimal presentation for these particular students and that the LC reduced the effects of this organization.

In conclusion, the study attempted to investigate the effects of differing aspects of learner control in the context of teaching APL. The overall outcome was that the learner control, as implemented, did not produce more effective learning and made no significant difference in student attitudes. This outcome is likely due to a number of factors which were not accounted for in the theoretical framework developed in Chapter III. Introducing learner control to a tutorial course is not likely to make much of an impact, particularly if the task or subject matter involves an inherently hierarchical structure or requires the development of problem solving skills. What is more likely to have an impact is an entirely different

instructional approach in which LC is an integral part, e.g., the problem solving laboratory style of BIP or the socratic style of SCHOLAR or SOPHIE. Individualization should be based upon a dynamically constructed model of the learners' skills and knowledge in terms of the subject matter. In other words, to exert a truly important effect on instruction, learner control implies a different type of interaction between the student and computer. The overall conclusion of this study, then, is that the potential benefits of learner control were likely overshadowed by the mismatch between the characteristics of the subject matter and learners and the general instructional approach. However, this conclusion can not be made directly from the results of this study and hence is a hypothesis for further research.

Suggestions for further research

Further evaluation of the value of LC in the teaching of APL should be conducted in the context of a more appropriate instructional strategy, i.e., a problem solving laboratory. LC should be an integral component of the instructional process, i.e., in the form of a command language used to direct the course of learning or as part of an intereactive dialogue with a socratic tutor. Based upon inferences about the student's learning styles and current APL knowledge (in terms of APL skills/facts) as well as inferences about misunderstandings based upon typical student errors, the system should attempt to guide the

student to mastery of APL skills and maximize performance. In doing this, the system would encourage patterns of LC which achieve this goal. However, since each student can be expected to make use of LC in different ways and to different degrees of effectiveness, this pattern of LC will be highly individual. Thus, some students may do best with control over a few instructional parameters while others may do best with a wide range of options. Furthermore, it may be most effective to provide the student with certain options in one part of the course (e.g., the beginning) and other options in another part of the course.

In essence this suggestion amounts to a recommendation for the study of how to properly balance student and computer control in an individualized manner. After a fairly complete review of LC research, Steinberg (1977) also recommended this direction for future LC studies. It is clear at this point that LC is not an instructional variable in itself, but rather a description of the interaction of other instructional variables (a higher order variable?) and needs to be studied as such.

TABLE 1

Studies of Learner Control in CAI from Judd et al. (1974).

Study	Subjects	Curriculum	LC Manipulation	Results
Grubb (1969)	50 IBM trainees	elementary statistics	sequence of chapters (within & between)	LC gp (within) scored better on postest
Dean (1969)	129 4-6th graders	arithmetic	control of amount of practice	LC gp scored better on postest
Newark (1970)	26 univ. students	comp. prog.	control of sequence and pacing	LC gp showed better retention, took less time
Barnes (1970)	214 3-13 graders	multiplication	selection of problem type	no differences
Olivier (1971)	176 univ. students	Xenograde systems	sequence of segments	LC gp did poorer than control
Judd et al. (1970)	univ. students	mathematics	sequence of segments	no differences
Judd et al. (1977)	univ. students	mathematics	amount of practice	LC gp worked more problems than controls
Gallager et al. (1970)	graduate students	?	sequence of segments	LC gp showed lower state anxiety levels
McCann et al. (1972)	U.S. Navy trainees	electronics	sequence of modules	no differences on performance, LC gp had more +ve attitudes

Table 2.
Comparison of Concept Sequencing from Three APL Texts.

IBM (1976)	Gilman & Rose (1970)	Gray (1973)
1.Using terminal	1.Using terminal	1.Using terminal
2.Arithmetic fns	2.Arithmetic fns	2.Variables
3.Variables	3.Exponentials	3.Aritmetic fns
4.Expressions	4.Logarithms	4.Logarithmic fns
5.Parentheses	5.Min/max	5.Exponential fns
6.List operations	6.Residue	6.Min/Max
7.List indexing	7.Relations	7.Residue
8.Min/Max	8.Variables	8.Arrays
9.Reduction	9.Reduction	9.Rank/Catenate
10.Tables (arrays)	10.Combinations	10.Indexing
11.Fns definition	11.Numerical fns	11.Fns definition
12.Fns editing	12.Fns editing	12.Fns editing
13.Local/global	13.Fns editing	13.local/global
14.Branching/loops	14.local/global	14.Branching/loops
15.Fnc I/O	15.Syst Cmds	15.Reduction
16.Catenation	16.Catenation	16.Inner/outer opr
17.Syst vars	17.Indexing, rank	17.Sorting fns
	18.Mixed fns	18.Permuting fns
	19.Encode/decode	19.Set fns
	20.Branching/loops	20.Encode/decode
	21.Arrays	21.Fnc I/O
	22.Transpose	22.Errors
	23.Inner/outer opr	23.Syst Cmds
	24.Fnc I/O	

Table 3.

Spiral Sequencing for Teaching APL.

Level I
- Variables and assignment (numerical)
- Data structures (scalars and vectors)
- Primitives (numerical, relations)
- Combinations (algorithms)
- Error messages

Level II
- Variables and assignment (literals, matrices)
- Data structures (matrices)
- Primitives (logical, selection, structural)
- Operators (sum/scan)
- Defined Functions (definition, editing)
- System commands
- Errors (workspace, defined fns)

Level III
- Primitives (circular, format, execute)
- Operators (inner/outer products)
- Defined Functions (branching, loops, I/O)
- System variables

Table 4.

APL Error Frequencies.

	Total Errors	Percentage
Function Execution	1131	91.8
VALUE	317	25.6
SYNTAX	314	25.4
DEFN	217	17.5
ENTRY	137	11.1
DOMAIN	50	4.0
LENGTH	34	2.7
RANK	34	2.7
INDEX	28	2.6
Workspace Management	106	8.2
INCORRECT COMMAND	52	4.2
WS NOT FOUND	26	2.1
NOT SAVED	9	0.7
NOT FOUND	7	0.5
WS FULL	5	0.4
STACK FULL	3	0.2
NOT COPIED	2	0.1
SYMBOL TABLE FULL	2	0.1

Table 5.

Attitude Questionnaire

1. I often feel frustrated while taking this course.
2. The material in this course is well organized.
3. It would have been better to learn this material in the usual classroom rather than via computer instruction.
4. I never find this course boring or dull.
5. Learning by the computer has a lot of disadvantages
6. I think the course should allow more student control and selection.
7. I am not learning as much from this course as I hoped I would.
8. I would like to take all of my instruction through the computer.
9. There are some aspects of APL that I would have liked to explore further if I had been able to.
10. Using the computer to learn allows you a lot of freedom in learning.

Table 6.

Use of LC Features.

	Total Accesses	Percentage
CONTROL FRAME		
Easy	82	9.3
Hard	148	16.8
Comment	29	3.3
Index	256	29.0
Examples	55	6.2
Next	267	30.3
Repeat	45	5.1
TOPIC INDEX		
All topics	54	15.0
Progress	195	54.4
Glossary	41	11.4
Return	69	19.2
PROGRESS FRAME		
Last concept	30	10.0
Topics done	72	24.0
Topics not done	107	35.7
Performance	91	30.3

Table 7.
Average Number of Accesses to the CONTROL Frame According to Course Segments.

		Segment							
		1	2	3	4	5	6	7	Avg.
LC/O	H.S. (N=6)	0.2	0.4	3.4	3.6	3.0	3.6	1.0	2.2
	Univ. (N=6)	0.7	1.4	9.4	1.6	3.3	1.9	0.6	2.7
LC/F	H.S. (N=7)	4.2	10.0	9.6	3.8	4.0	4.8	3.8	5.7
	Univ. (N=5)	4.9	7.0	7.0	1.7	1.3	1.1	0.7	3.4

Table 8.

Total Average Correct According to LC Conditions for Each Segment of APLLC.

		Segments							
		1	2	3	4	5	6	7	Total
Univ	No-LC (N=5)	11.2	12.6	83.0	8.5	12.0	7.7	7.0	127
	LC/O (N=6)	11.2	17.0	82.8	9.5	10.7	12.5	6.3	120
	LC/F (N=5)	8.6	67.8	64.2	13.0	11.0	8.0	8.0	81.4
H.S.	No-LC (N=7)	15.1	26.4	74.3	7.0	9.3	6.0	--	122
	LC/O (N=6)	12.5	17.0	59.0	6.4	7.8	10.6	5.0	91
	LC/F (N=5)	5.4	22.0	78.8	7.3	7.7	17.0	4.5	87

Table 9.
Results of ANOVA for Performance Measures.

Performance Measures	d.f.	Univ. F	H.S. F
Completion times	2	0.10	0.27
Topics	7	32.7	14.9
Interaction	14	0.32	1.26
Total correct	2	0.63	0.04
Topics	7	32.43	19.5
Interaction	14	0.85	1.93
Time/Correct ratio	2	0.58	1.25
Topics	7	22.6	20.7
Interaction	14	1.63	0.57

Table 10.
Average Rating of Attitude Items According to LC Conditions (1=agree; 5=disagree).

		Items										
		1	2	3	4	5	6	7	8	9	10	Comp.
No-LC	Univ. (N=3)	3.3 (0.6)	3.3 (1.2)	3.7 (0.6)	1.7 (1.1)	3.3 (1.2)	3.0 (1.0)	3.3 (1.5)	1.3 (0.6)	2.3 (1.5)	4.3 (0.6)	28 (1.1)
	H.S. (N=3)	2.7 (0.6)	5.0 (0.0)	2.7 (0.6)	2.0 (1.0)	2.7 (1.5)	2.3 (1.2)	3.0 (1.0)	3.0 (2.0)	4.0 (1.0)	3.3 (1.5)	32 (3.6)
LC/O	Univ. (N=6)	2.5 (1.3)	3.2 (1.2)	3.7 (0.8)	2.8 (1.3)	3.5 (1.0)	2.5 (1.4)	3.2 (1.5)	2.8 (1.2)	2.3 (1.8)	3.2 (1.0)	30 (2.6)
	H.S. (N=3)	3.3 (0.6)	2.7 (0.6)	3.0 (1.0)	1.3 (0.6)	3.7 (1.5)	3.3 (1.2)	4.3 (1.2)	2.3 (1.2)	3.0 (1.0)	3.3 (2.1)	25 (5.3)
LC/F	Univ. (N=3)	2.3 (1.2)	3.7 (1.5)	3.3 (2.1)	3.7 (1.5)	3.7 (1.5)	2.0 (0.0)	2.0 (1.7)	2.0 (1.0)	2.3 (1.5)	4.0 (1.0)	33 (5.0)
	H.S. (N=3)	2.3 (0.6)	3.7 (1.5)	3.3 (2.1)	3.0 (2.0)	3.7 (1.5)	3.3 (1.5)	2.3 (1.5)	1.3 (0.6)	4.0 (0.0)	4.3 (0.6)	29 (5.5)

Note: Standard deviations are in brackets below means.

Table 11.
Average Attitude Ratings for High and Low Scores on Mathematical and Programming Measures and Internal/External Locus of Control Scale (1=agree; 5=disagree).

	Items 1	2	3	4	5	6	7	8	9	10	Comp.
Programming Level											
High (N=8)	3.0	4.0	3.6	3.0	3.4	2.5	3.5	2.7	2.9	4.1	31.0
	(0.7)	(1.1)	(0.9)	(1.8)	(1.3)	(1.1)	(1.5)	(1.6)	(1.4)	(0.6)	(3.6)
Low (N=13)	2.5	3.2	3.2	2.2	3.5	2.8	2.8	1.9	2.9	3.4	29.0
	(1.1)	(1.2)	(1.4)	(1.0)	(1.2)	(1.2)	(1.2)	(0.9)	(1.5)	(1.5)	(4.6)
Mathematical Level											
High (N=9)	2.8	3.4	3.6	2.4	3.1	2.7	3.5	2.6	2.8	3.8	30.0
	(0.7)	(1.2)	(0.9)	(1.7)	(1.1)	(1.0)	(1.1)	(1.2)	(1.5)	(0.8)	(3.0)
Low (N=12)	2.6	3.7	3.0	2.6	3.8	2.7	2.5	1.6	3.0	3.5	30.0
	(0.9)	(1.2)	(1.4)	(1.2)	(1.1)	(1.3)	(1.4)	(0.8)	(1.4)	(1.4)	(5.1)
Internal/External Locus of Control											
High (N=9)	2.4	3.9	3.6	2.9	3.7	2.4	2.8	1.7	3.2	3.6	30.0
	(0.9)	(1.2)	(0.8)	(1.5)	(0.9)	(1.1)	(1.3)	(0.7)	(1.1)	(1.1)	(3.1)
Low (N=12)	3.0	3.2	3.1	2.1	3.2	3.0	3.3	2.7	2.6	3.7	30.0
	(1.0)	(1.2)	(1.2)	(1.3)	(1.3)	(1.1)	(1.4)	(1.3)	(1.5)	(1.3)	(5.0)

Note: Standard deviations are in brackets below means.

Table 12.
Means and Standard Deviations of the Scores on the Programming Quizzes for the University Students.

	Quiz 1		Quiz 2	
	Mean	S.D.	Mean	S.D.
No-LC (N=4)	3.8	2.5	4.5	0.6
LC/O (N=4)	1.8	2.0	1.5	1.7
LC/F (N=4)	2.5	2.0	3.5	1.3

Table 13.
Correlations Between Scores on Quizzes and Performance or Attitudinal Measures from APLLC. (N=16)

	Quiz 1	Quiz 2
Total time	-.60	-.69
Total correct	.16	.05
Total Correct/time	.61	.40
CAI/LC attitude (composite)	-.22	-.48

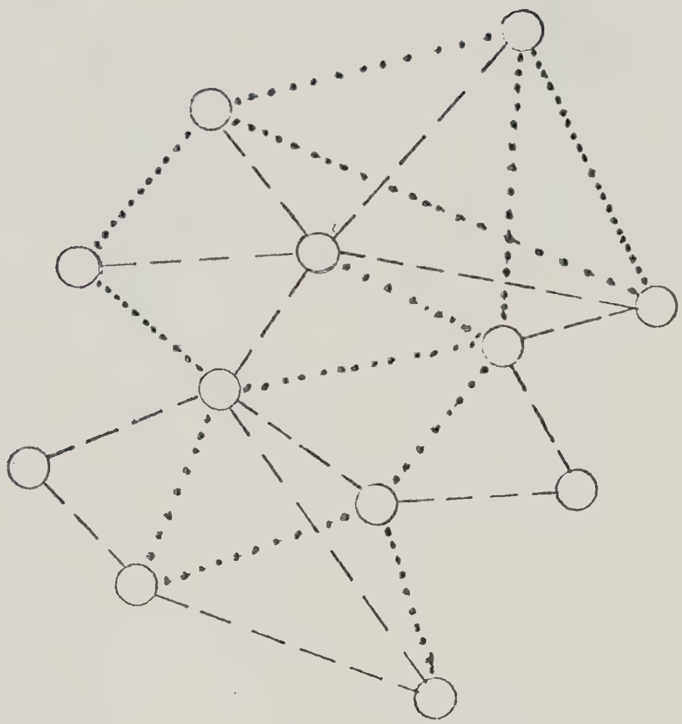
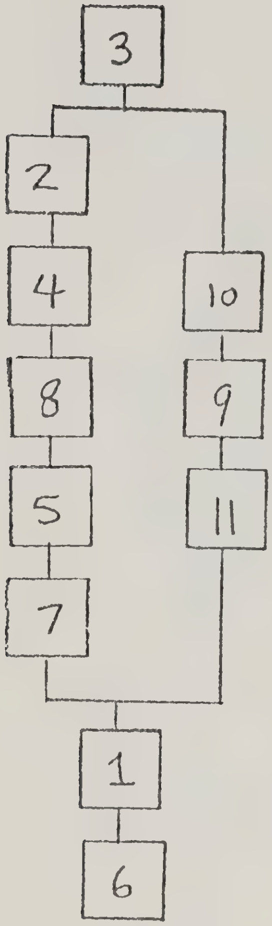


FIGURE 1

Network representation showing two different subgraphs which contain the same nodes with different interrelationships.



CONCEPTS

1. Assignment to variables
2. Comparative signs
3. Compression
4. Exponentiation
5. Max/Min
6. Parentheses
7. Right to Left Rule
8. Residue
9. Take & Drop
10. Vector Arithmetic
11. Vector Indexing

Figure 2. Task Structure for 11 APL Concepts from O'Neal (1977).

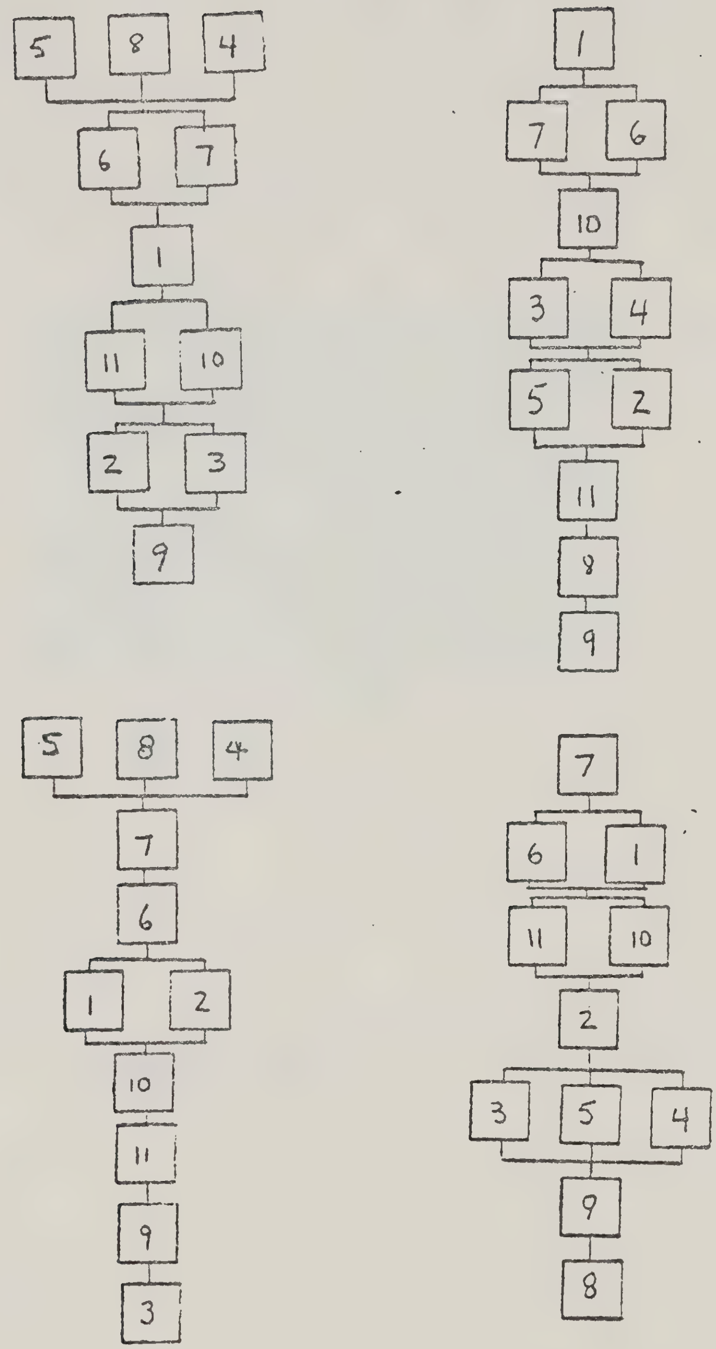


Figure 3. Task structures generated by 4 APL experts.

```
    ∇R←A FIB

[1] R←B/A
[2] ∇

    A 0
SYNTAX ERROR
A[1] R←B/A
         ∧

    FIB 0
SYNTAX ERROR
FIB 0
  ∧

    A 1 2 3 4 5 6 7
SYNTAX ERROR
A[1] R←B/A
         ∧
```

Figure 4. An example of a misunderstanding of function arguments in APL.

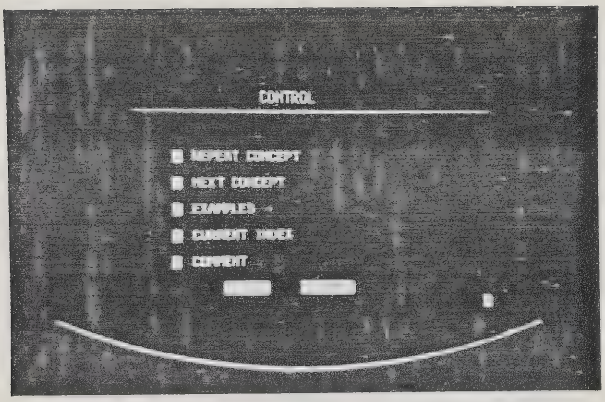


Figure 5. CONTROL display in APLLC.

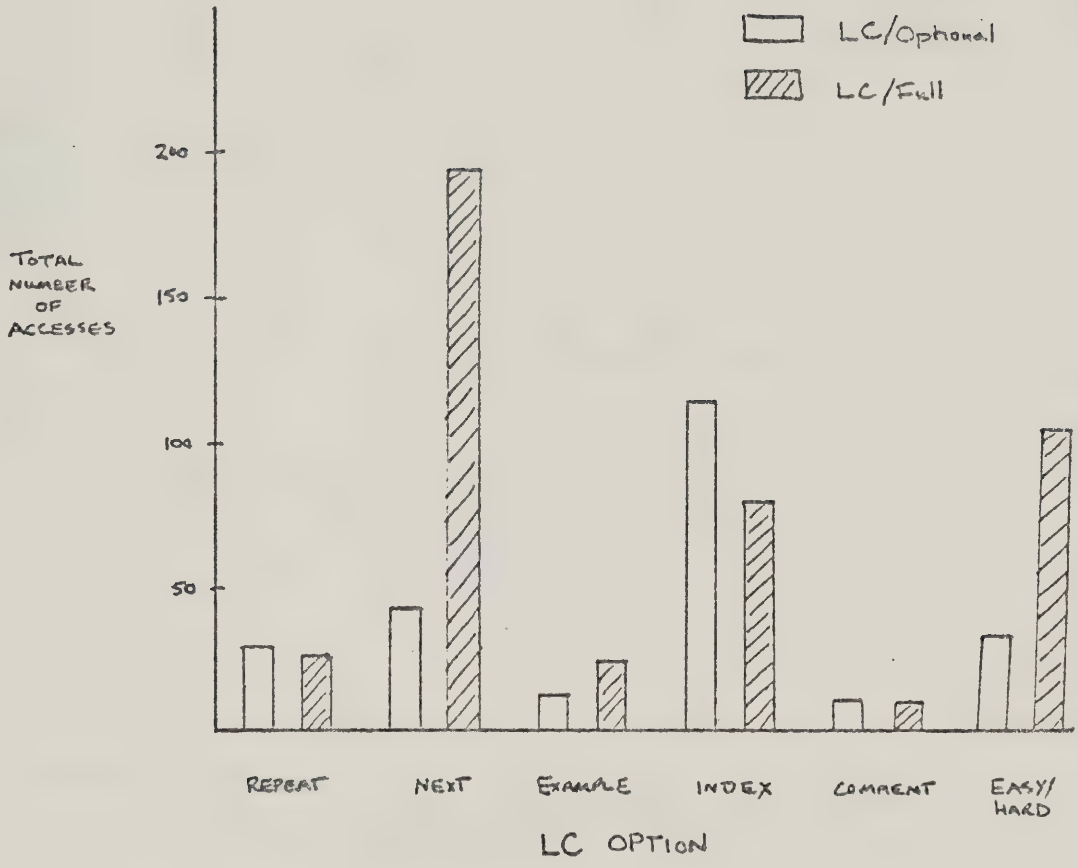


Figure 6. Use of LC Options for Optional and Full LC conditions pooled across both student groups.

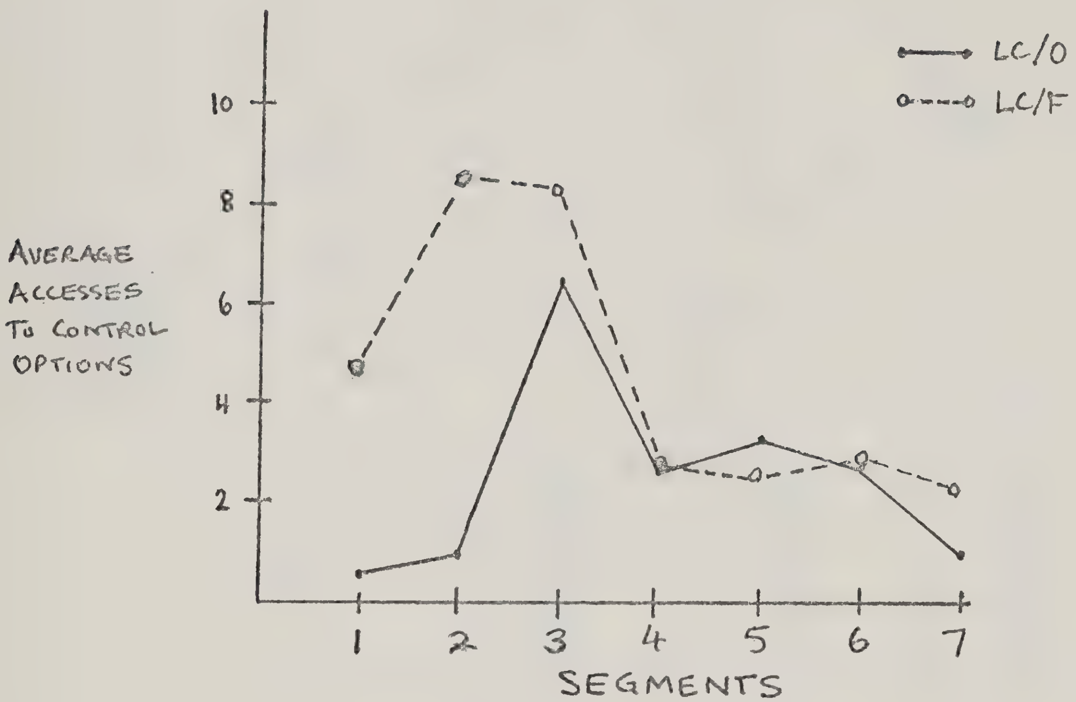


Figure 7. Average number of accesses to CONTROL options according to segments (pooled across both student groups).

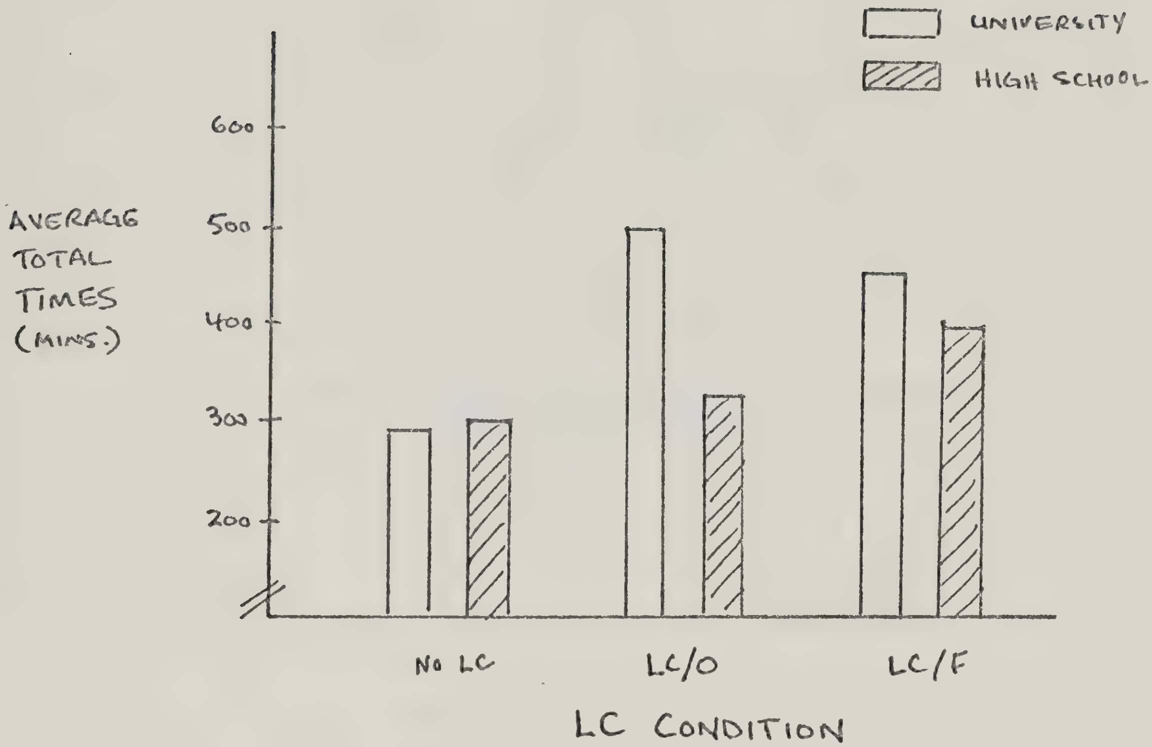


Figure 8. Average Total Completion Times according to LC Conditions and student group.

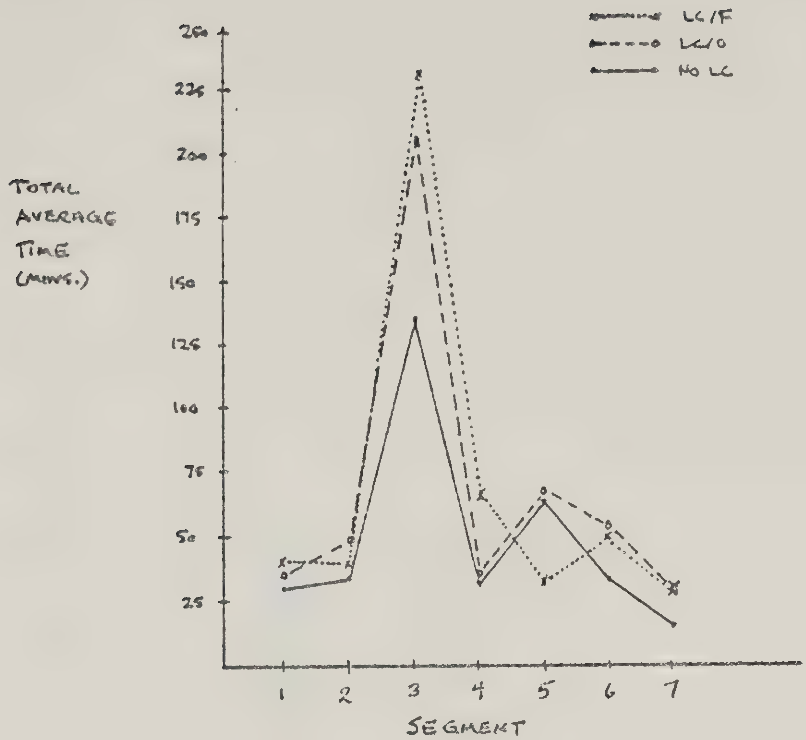


Figure 9a. Average Total Completion Time by Segments for the University students.

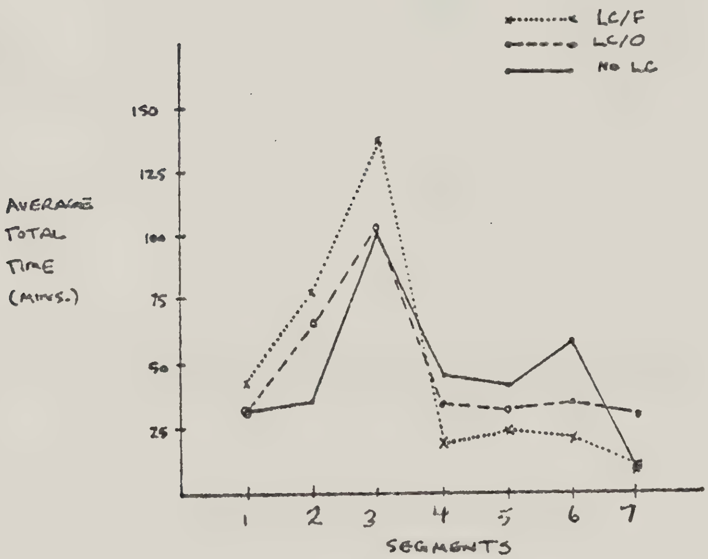


Figure 9b. Average Total Completion Time by Segments for the High School student group.

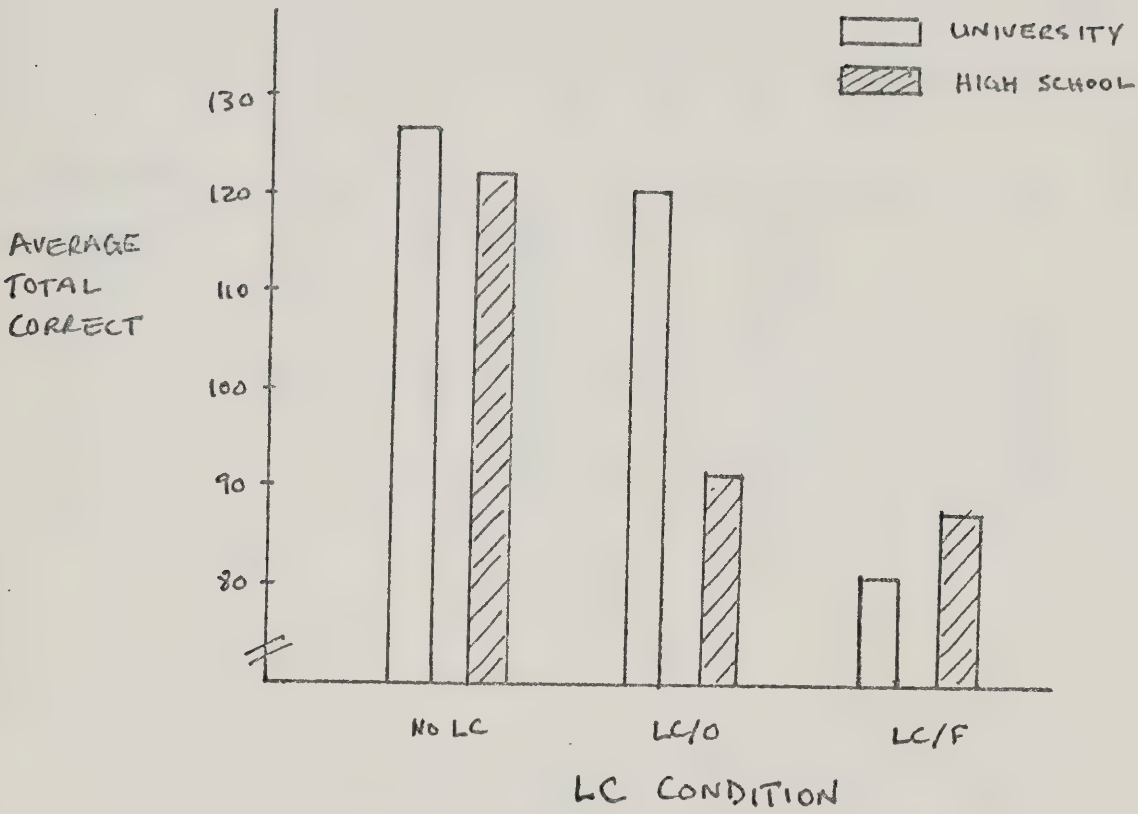


Figure 10. Average Total Number Correct according to LC Conditions and student groups.

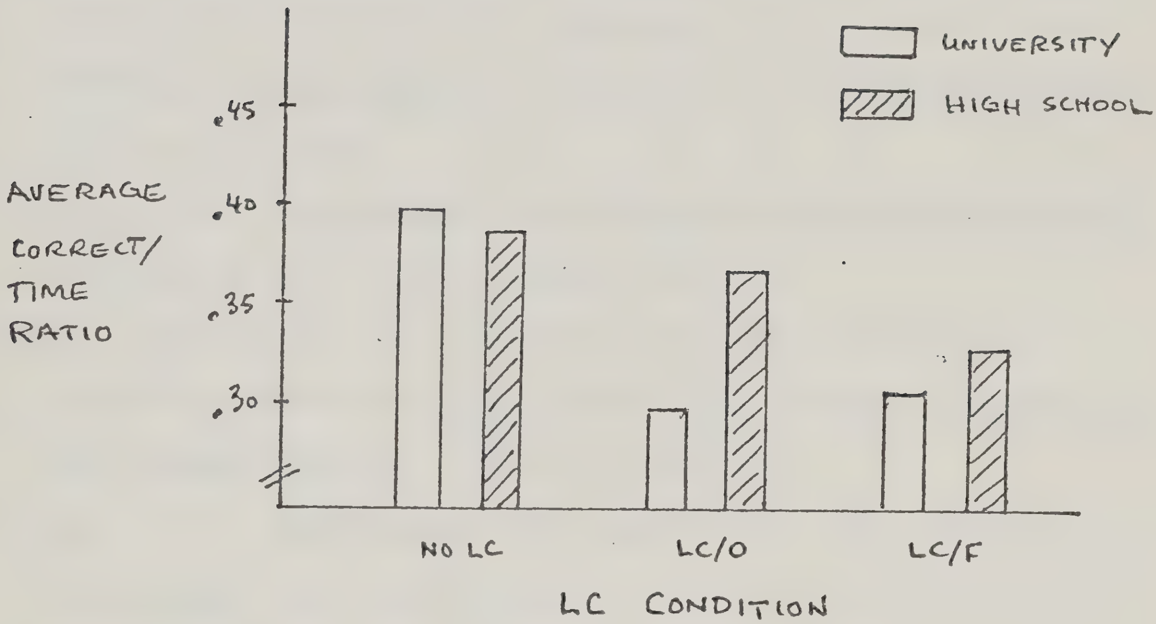


Figure 11. Average Correct/Time Ratio according to LC Condition and student group.

VI. REFERENCES

Atkinson, R.C. Optimizing the learning of a second language vocabulary. Journal of Experimental Psychology , 1972, 96 , 124-129.

Atkinson, R.C. Adaptive instructional systems: Some attempts to optimize the learning process. In D. Klahr (ed.), Cognition & Instruction. Hillsdale, N.J.: LEA Publishers, 1976. pp. 81-108.

Ausubel, D.P. Educational Psychology: A cognitive view. New York: Holt, Rinehart & Winston, 1968.

Barr, A., Beard, M., & Atkinson, R.C. The computer as a tutorial laboratory: The Stanford BIP project. Journal of Man-Machine Studies , 1976, 8 , 567-596.

Briggs, L.J. Sequencing of instruction in relation to hierarchies of competence. American Institute for Research Report No. 3. Pittsburgh, Penn, 1968.

Brown, J.S. & Burton, R.R. Multiple representations of knowledge for tutorial reasoning. In D.G. Bobrow & A. Collins (Eds.), Representation and understanding , New York: Academic Press, 1975.

Bruner, J.S. Toward a theory of instruction. Cambridge, Mass.: Belnap Press, 1966.

Bunderson, C.V. The design and production of learner-controlled courseware for the TICCIT system: A progress report. Man-Machine Studies , 1974, 6 , 479-491.

Campbell, V.N. Self direction and programmed learning for

five different types of learning. *Psychology in the Schools*, 1964, *1*, 348-359.

Carroll, J.B. Psychometric tests as cognitive tasks: A new structure of intellect. In L. Resnick (Ed.), *The nature of intelligence.*, Hillsdale, N.J.: LEA Publishers, 1976.

Collins, A., Warnock, E.H., Aiello, N., & Miller, M.L. Reasoning from incomplete knowledge. In D.G. Bobrow & A. Collins (Eds.), *Representation and understanding*, New York: Academic Press, 1975.

Cronbach, L.J. & Snow, R.E. *Aptitudes and instructional methods.* New York: Irvington Press, 1977.

Dwyer, T.A. Heuristic strategies for using computers to enrich education. *Man-Machine Studies*, 1974, *6*, 137-154.

Fisher, M.D., Blackwell, L.R., Garcia, A.B., & Greene, J.C. Effects of student control and choice on engagement in a CAI arithmetic task in a low income school. *Journal of Educational Psychology*, 1975, *67*, 776-783.

Fredericks, P.S. *The effects of locus of control on CAI performance.* AERA Annual meeting, San Francisco, April 1976.

Friere, P. *Pedagogy of the oppressed.* Penguin Books:England, 1972.

Gagne, R.M. Learning hierarchies. *Educational Psychologist*, 1968, *6*, 1-9.

Gagne, R.M., & Briggs, L.J. *Principles of instructional design.* New York: Holt, Rinehart & Winston, 1974.

Geis, G.L. Student participation in instruction. Journal of Higher Education , 1976 47 , 249-273.

George, T.W. Student-control of learning: A review of research. Programmed Learning & Educational Technology , 1976, 13 , 55-57.

Gilman, L & Rose, A.J. APL/360: An interactive approach. New York: Wiley, 1970.

Gray, L.D. A Course in APL/360 with applications. 1973.

Hunt, E.B., Frost, N., & Lunneborg, C. Individual differences in cognition: A new approach to intelligence. In G. Bower (Ed.), Advances in Learning & Motivation, Vol. 7. , New York: Academic Press, 1974.

IBM Corporation Independent Study Guide for APL/VS. White Plains, N.Y., 1976.

Illich, I. Deschooling society New York: Harper & Row, 1970.

Issing, L.J., & Eckert, A. Teacher controlled versus learner controlled instruction - a comparative study. British Journal of Educational Technology , 1973, 4 , 149-154.

Iverson, K. A programming language. New York: Wiley, 1962.

Judd, W.A., O'Neil, H.F., & Spelt, P.F. Individual differences and learner control I: Program development and investigation of control over mnemonics in computer-assisted instruction. AFHRL-TR-74-3, July 1974.

Lahey, G.F. Learner control of Computer based instruction: A comparison with guided instruction. Proceeding of the 1978 ADCIS Annual Conference. Dallas, Texas, March 1978.

Lahey, G.F., & Crawford, A.M. Learner control of lesson

strategy: some tentative results. Paper presented at AERA Conference, San Francisco, 1976.

Landa, L. *Algorithmization in learning and instruction.* Englewood Cliffs, N.J.: Educational Technology Publishers, 1974.

Landa, L. Cybernetic Methods in Education. *Educational Technology*, 1977, *17(10)*, 7-12.

Lefcourt, H.M., Sordoni, C. & Sordoni, C. Locus of control and the expression of humor. *Journal of Personality*, 1974, *42*, 130-143.

Luehrmann, A.W. Should the computer teach the student or vice-versa? *Proceedings of the Spring Joint Computer Conference*, 1972. Reprinted in *Creative Computing*, November 1976, p 42-45.

Mager, R.F. & Clark, C. Explorations in student controlled instruction. *Psychological Reports*, 1963, *13*, 71-76.

Mayer, R.E. The sequencing of instruction and the concept of assimilation-to-schema. *Instructional Science*, 1977, *6*, 367-388

McGee, P.E., & Crandall, V.C. Beliefs in internal-external control of reinforcements and academic performance. *Child Development*, 1968, *39*, 91-102.

Merrill, M.D. Learner Control: Beyond aptitude-treatment interactions. *Audio-Visual Communications Review*, 1975, *23*, 217-226.

Norman, D.A. *Studies of learning and self contained educational systems.* Center for Human Information

Processing report 7601, University of California at San Diego, 1976.

O'Neal, A.F. The effect of test qualification criteria on a learner controlled hierarchical rule-using task. PhD Thesis, Brigham Young University, April 1977.

Papert, S. & Solomon, C. Twenty things to do with a computer. Educational Technology, 1972, 12, 9-16.

Pask, G. Conversation, Cognition and Learning. New York: Elsevier, 1975.

Pask, G. Styles and strategies of learning. British Journal of Educational Psychology, 1976, 46, 128-148.

Pask, G. & Scott, B.C.E. Learning strategies and individual competence. Man-Machine Studies, 1972, 4, 217-253.

Peele, H.A. Computer glass boxes: Teaching children concepts with "A Programming Language." Educational Technology, 1974, 14, 9-16.

Popham, W.J. Educational evaluation. Englewood Cliffs, N.J.: Prentice-Hall, 1975.

Posner, G.J. & Strike, K.A. A categorization scheme for principles of sequencing content. Review of Educational Research, 1976, 46 (1), 665-690.

Rogers, C. Freedom to learn. Columbus, Ohio: Merrill Publishing, 1969.

Rotter, J.B. Generalized expectancies for for internal versus external control of reinforcement. Psychological Monographs, 1966, 80, Whole No. 609.

Saal, H.J. & Weiss, Z. An empirical study of APL programs.

Computer Languages , 1977, 2 p. 47-59.

Scandura, J. What is a rule? Journal of Educational Psychology , 1972, 63 , 179-185.

Scandura, J.M. Structural approach to instructional problems. American Psychologist , 1977, 32 , 33-53.

Segal, E.M. & Stacey, E.M. Rule governed behavior as a psychological process. American Psychologist , 1975, 30 , 541-552.

Seidel, R.J., Wagner, H., Rosenblatt, R.D., Hillelsohn, M., & Stelzer, J. Learner control of instructional sequencing within an adaptive tutorial CAI environment. Instructional Science , 1978, 7 , 37-80.

Self, J.A. Student models in computer assisted instruction. Journal of Man-Machine Studies , 1974, 6 , 261-276.

Skinner, B.F. The technology of teaching. New York: Appleton-Century-Crofts, 1968.

Steinberg, E.R. Review of student control in computer-assisted instruction. Journal of Computer Based Instruction , 1977, 3 , 84-90.

Stolurow, L.M. Application of psychology to educational technology. Educational Technology , 1972, 12 ,9-14.

Suppes, P. & Morningstar, M. Computer assisted instruction at Stanford, 1966-1968. New York: Academic Press, 1972.

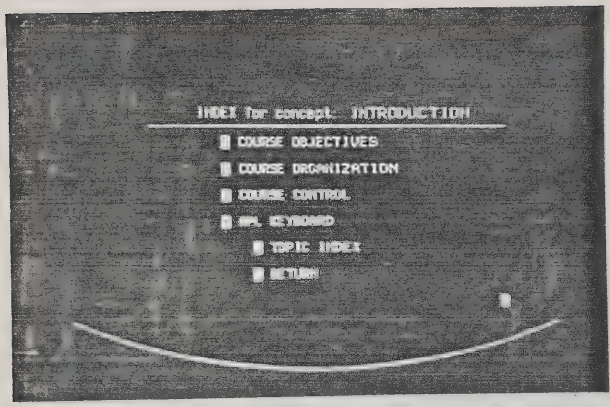
Tennyson, R.D. A review of experimental methodology in instructional task sequencing. AV Communications Review , 1972, 20 , 147-159.

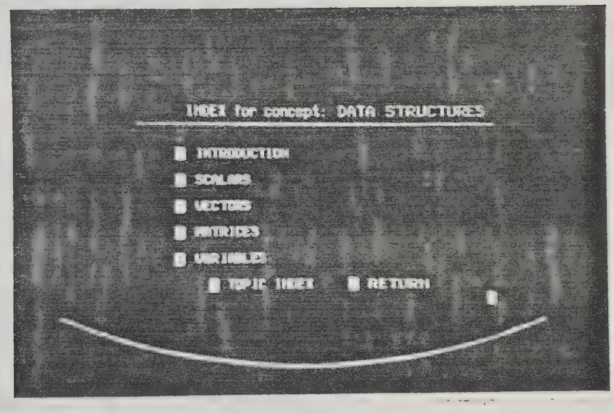
White, A.L. & Smith, D.D. A study of the effects of varying

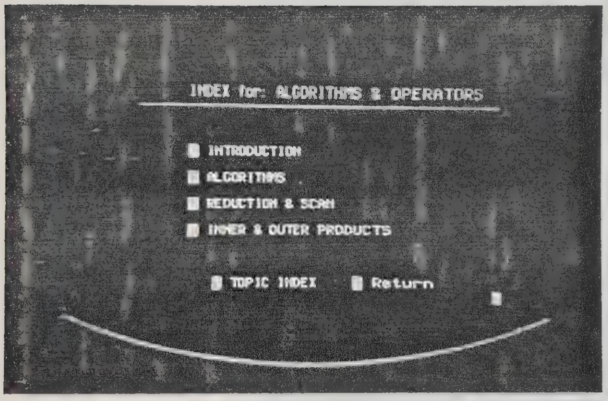
student responsibility for instructional descisions in a CAI course. _Journal of Computer Based Instruction_ , 1974, _1_ , 13-21.

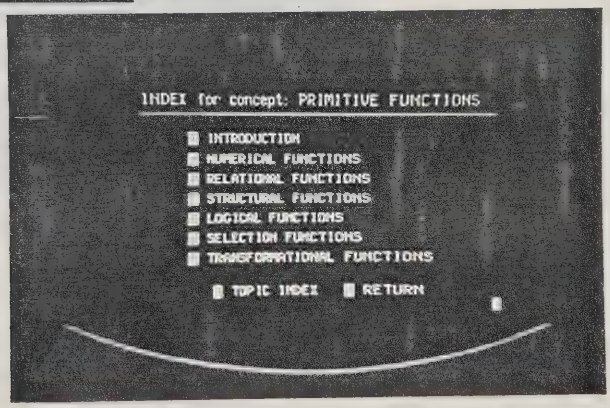
VII. APPENDIX I

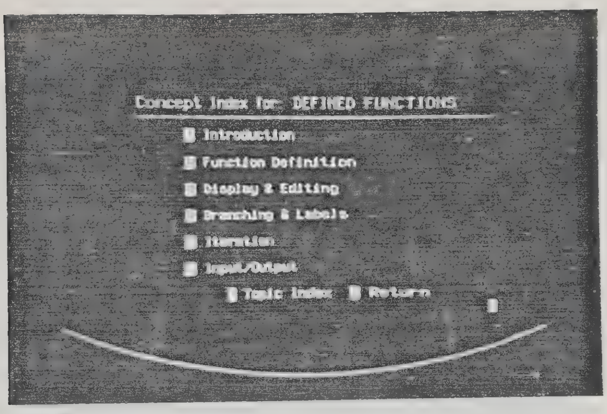
The following photographs of screen displays illustrate the concept indicies for the 7 topics in APLLC, the topic index, the progress record menu, and the nature of the performance summary available.

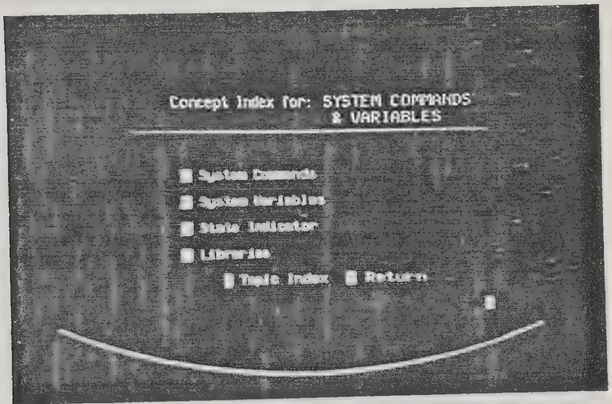

INDEX for concept: INTRODUCTION
COURSE OBJECTIVES
COURSE ORGANIZATION
COURSE CONTROL
APL KEYBOARD
TOPIC INDEX
RETURN

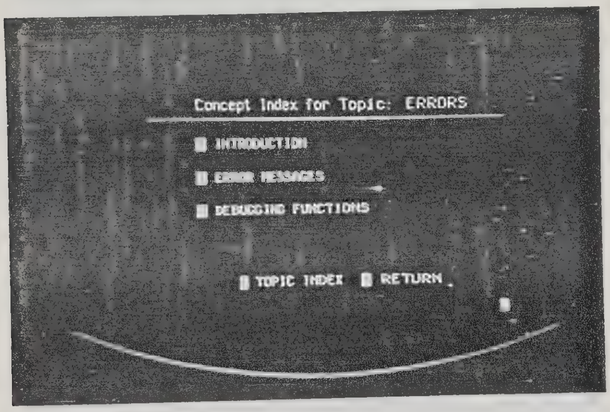

INDEX for concept: DATA STRUCTURES
INTRODUCTION
SCALARS
VECTORS
MATRICES
VARIABLES
TOPIC INDEX
RETURN

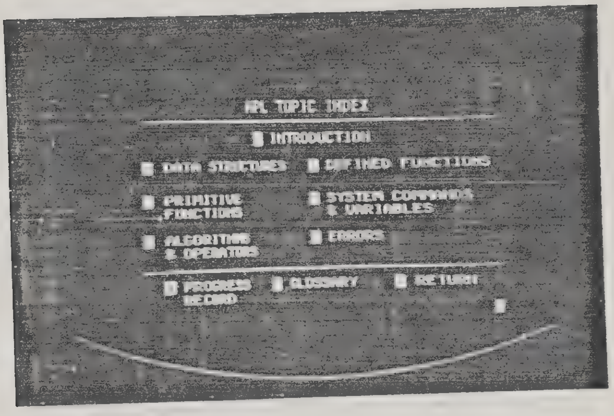

INDEX for: ALGORITHMS & OPERATORS
INTRODUCTION
ALGORITHMS
REDUCTION & SCAN
INNER & OUTER PRODUCTS
TOPIC INDEX
Return

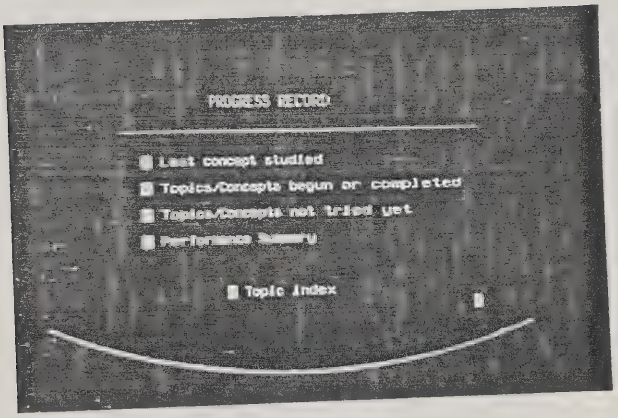

INDEX for concept: PRIMITIVE FUNCTIONS
INTRODUCTION
NUMERICAL FUNCTIONS
RELATIONAL FUNCTIONS
STRUCTURAL FUNCTIONS
LOGICAL FUNCTIONS
SELECTION FUNCTIONS
TRANSFORMATIONAL FUNCTIONS
TOPIC INDEX
RETURN

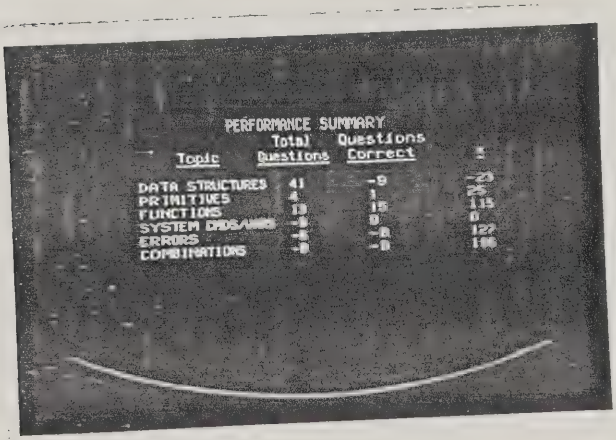

Concept Index for: DEFINED FUNCTIONS
Introduction
Function Definition
Display & Editing
Branching & Labels
Iteration
Input/Output
Topic Index
Return


Concept Index for: SYSTEM COMMANDS
& VARIABLES
System Commands
System Variables
State Indicator
Libraries
Topic Index
Return


Concept Index for Topic: ERRORS
INTRODUCTION
ERROR MESSAGES
DEBUGGING FUNCTIONS
TOPIC INDEX
RETURN


APL TOPIC INDEX
INTRODUCTION
DATA STRUCTURES
DEFINED FUNCTIONS
PRIMITIVE FUNCTIONS
SYSTEM COMMANDS & VARIABLES
ALGORITHMS & OPERATORS
ERRORS
PROGRESS RECORD
GLOSSARY
RETURN


PROGRESS RECORD
Last concept studied
Topics/Concepts begun or completed
Topics/Concepts not tried yet
Performance Summary
Topic index


PERFORMANCE SUMMARY
Total Questions
Questions Correct
Topic
DATA STRUCTURES
PRIMITIVES
FUNCTIONS
SYSTEM
ERRORS
COMBINATIONS

B30226